الجمهورية اللبنانية
وزارة التربية والتعليم العالي
المديرية العامة للتعليم المهني والتقني

قـرار رقـم ١٤٨ /٢٠٢٤

إعتماد المنهاج الرسمي لإختصاص " تكنولوجيا المعلوماتية" المعدل
لمستوى شهادة البكالوريا الفنية

إن المديرة العامة للتعليم المهني والتقني بالتكليف،
بناء على المذكرة الإدارية رقم 65/م/2019 تاريخ 2019/5/24 (قبول طلب المدير العام بالتكليف للتعليم المهني والتقني إعفاء من تكليفه بهذه المديرية العامة وتكليف مدير المعهد الوطني للعناية التمريضية بمهام المدير العام لها)،
بناء على المرسوم رقم 7880 تاريخ 1967/7/25 وتعديلاته (تنظيم حقول ومراحل وشهادات التعليم المهني والتقني)،
بناء على المرسوم رقم 8590 تاريخ 2012/8/2 (تنظيم حقول ومراحل وشهادات التعليم المهني والتقني)،
إستناداً الى مذكرة التفاهم الموقعة بين جمعية بنات مريم أم المعونة والمديرية العامة للتعليم المهني والتقني بتاريخ 2021/5/18 (التعليم التقني والمهني لتحسين الكفاءات وفرص العمل اللائق والإزدهار للشباب المستضعفين في لبنان)
إستناداً الى الدراسة التي انجزت في مراجعة وتعديل منهاج إختصاص "تكنولوجيا المعلوماتية" بين خبراء في جمعية بنات مريم أم المعونة وفريق من الأساتذة في المديرية العامة للتعليم المهني والتقني،
بناء على إقتراح رئيس المصلحة الفنية بالتكليف،

يقرر ما يأتي:

المادة الاولى: يُعتمد المنهاج الرسمي المعدل لإختصاص " **تكنولوجيا المعلوماتية**" لمستوى شهادة البكالوريا الفنية، في معاهد ومدارس التعليم المهني والتقني الرسمية والخاصة وفقاً للمنهاج الملحق بهذا القرار إعتباراً من العام الدراسي 2024-2025:

المادة الثانية: يُبلغ هذا القرار من يلزم، وينشر على الموقع الإلكتروني للمديرية العامة للتعليم المهني والتقني.

الدكوانه في: ١٢ آذار ٢٠٢٤
المديرة العامة للتعليم المهني والتقني بالتكليف

د.هنادي بري

صورة طبق الأصل
مروه نابلسي

Technical and Vocational Education to Improve Competences and Opportunities

for Decent Work and Prosperity for Young Vulnerable People in Lebanon

AID 012590/02/0

CURRICULUM FOR BACCALAURÉAT TECHNIQUE

SUBJECT "INFORMATION TECHNOLOGY"

21 March 2024

In partnership with

DISCLAIMER

The AICS-funded project EDU TOP - Technical and Vocational Education to Improve Competences and Opportunities for Decent Work and Prosperity for Young Vulnerable People in Lebanon (AID 012590/02/0) has developed this Curriculum in the framework of its overall effort to promote a technical education system of quality, equitable and inclusive, responsive to the needs of the labor market, in favor of the most excluded communities and individuals.

This Curriculum was updated by *COMI ETS - Cooperazione per il Mondo in via di Sviluppo* and its local counterpart *FMA - Filles de Marie Auxiliatrice*, in coordination with the *DGTVET - Directorate General of Technical and Vocational Education and Training* and with the support of *Fondazione CIOFS-FP ER EMILIA ROMAGNA ETS* and *A.E.M.S. S.A.L. Advance Engineering and Management Services.*

This Curriculum was produced with the contribution of AICS - Italian Agency for Development Cooperation. Its contents are exclusive responsibility of COMI ETS and do not necessarily represent the views of the Agency.

IT Technology

At the end of their training, the BT student will be capable to perform the following tasks:

- Assembling computers
- Installation, maintenance and diagnostic of computers
- IT fundamentals (operating system, office, e-mail, web browsing….)
- Configurations and maintenance of mobile smart devices
- Programming
- Practice of local networks
- Basic web page design
- Basics of Cloud computing
- Storing data (SQL, No SQL, cloud)
- IT security

SUBJECT	1st YEAR		2nd YEAR		3rd YEAR	
	course	**lab**	**course**	**lab**	**course**	**lab**
Arab	60		60		60	
French	60		60		60	
English	90		90		90	
Social Sciences	30		30		30	
General mathematics	120		120		120	
Right	30		30		60	
TOTAL	**390**		**390**		**420**	
Sporting activities	30		30		30	
TOTAL	**30**		**30**		**30**	
Introduction to algorithms	60					
Computer technology I / II / III	60	60		30		30
Applied Computing I / II		120		60		
Introduction to electricity and electronics		30				
Smart mobile device I / II		60		30		
Computer Assembly I / II		30		30		
Programming I / II // III	30	60	30	60	30	60
Programming the web I / II / III		30		60		60
Networks and Security Protocols I / II // III	30		60	60	30	90
Maintenance methodology (incl. 2 weeks in company)					30	60
Basics of cloud computing and AI I / II			30	30	30	60
Storing Data I / II			60	30	30	30
Total	**180**	**390**	**180**	**390**	**150**	**390**
TOTAL	**990**		**990**		**990**	

FIRST YEAR

SPECIALIZATION SUBJECTS

1. INTRODUCTION TO ALGORITHMS
2. COMPUTER TECHNOLOGY I
3. APPLIED COMPUTING I
4. INTRODUCTION TO ELECTRICITY AND ELECTRONICS
5. SMART MOBILE DEVICE I
6. COMPUTER ASSEMBLY I
7. PROGRAMMING I
8. PROGRAMMING THE WEB I
9. NETWORKS AND SECURITY PROTOCOLS I

INTRODUCTION TO ALGORITHMS
(60 PERIODS)

OBJECTIVE

At the end of this subject, the student should be able to:

- understand how to break down a problem to better solve it.
- to be able to go from a problem to its algorithmic solution.
- establish a simple, fair, efficient and fast algorithm.

CHAPTER 1 (4 periods)

General Presentation of Algorithms

1.1 Introduction. overview of what an algorithm is and its importance
1.2 Basic concepts and terminology in algorithms
1.3 The stages of implementing an application (examples)

CHAPTER 2 (4 periods)

Construction of algorithms

2.1 Introduction
2.2 The algorithmic file
2.3 Structure of the environment
2.4 Example of an algorithmic file
2.5 Organization charts
2.6 Trace of the algorithm
2.7 Remarks

CHAPTER 3 (16 periods)

General introduction to algorithms

3.1 Definition and basic concepts
3.2 Constants
3.3 The notion of variable
3.4 Basic types
3.5 Operators (logical, arithmetic, relationship, …)
3.6 Expressions (logical, arithmetic, etc.)
3.7 Actions (input and output)
3.8 Parameters (formal and effective)
3.9 Examples and exercises

CHAPTER 4 (20 periods)

Structure of an algorithm

4.1 Conditional algorithms
  4.1.1 conditional scheme (if … then … finish)
  4.1.2 alternatives (if … then … else … finish).
  4.1.3 nested schemas.
4.2 Selection and choice (cases)
4.3 Iterative algorithms

4.3.1 general diagram
4.3.2 pattern repeat
4.3.3 schema while
4.3.4 diagram for
4.3.5 nested loops

4.4 Examples and exercises (for all exercises the student must draw the flow charts/traces)

CHAPTER 5 (6 periods)

Understanding problem solving and algorithmic thinking

5.1 Techniques for breaking down problems
5.2 Case studies of real-world problems and how they can be solved algorithmically
5.3 Introduction to basic problem-solving strategies (e.g., divide and conquer, greedy algorithms)

CHAPTER 6 (6 periods)

Arrays (notions)

6.1 One-dimensional arrays
6.1.1 definitions
6.1.2 declaration
6.1.3 designation of an element
6.1.4 browse and sequential search
6.1.5 examples and exercises

CHAPTER 7 (4 periods)

Introduction to current use cases

7.1 overview of current trends including machine learning, ai, web development
7.2 discussion on how algorithms are integral to these technologies
Examples should include:
-web: search algorithms, sorting algorithms for e-commerce, and content ranking
-mobile: how algorithms drive the functionality of mobile apps, including gps navigation, fitness trackers, and gaming apps
7.3 ethical considerations (including bias in algorithms and privacy concerns)

COMPUTER TECHNOLOGY I
(120 PERIODS OF WHICH 60 ARE PRACTICAL EXERCISES)

OBJECTIVE

The objective of this course is to provide students with a general introduction to computer hardware and operating systems.

This presentation covers the overall functioning of the computer, its main functional units, and key components (processor, memory, motherboard, peripherals).

For each component, its general structure, functioning, and features should be presented.

Once the hardware topics are completed, students will also see how operating systems integrate with the hardware.

This program aims to balance theory and practice to ensure that students gain a comprehensive understanding of the internal workings of a computer.

However, it is recommended to adjust the distribution of hours based on the specific needs of students and the desired depth of study.

INSTRUCTIONS FOR TEACHERS

The main concepts of this subject will be further elaborated in BT2 and BT3 through more specific practical exercises.

During the scheduled hours in BT1, students should become familiar with the different components of a computer, understanding the function and main features of each component, as well as the most common models and brands on the market.

The practical laboratory hours will be structured to carry out exercises on the theoretical aspects covered and to analyze, in real-time, the various components studied during the theoretical hours.

Finally, the laboratory hours include the completion of a final project (8 periods) where students will apply the knowledge acquired during the course in a practical manner. For example, this could involve a detailed analysis of a hardware component, solving a complex problem, or the advanced configuration of a system.

CHAPTER 1 (5 periods)

General presentation of the computer

1.1 What is a computer used for?
1.2 Introduction and Contextualization in Industry 4.0
1.3 Definition of a computer
1.4 The main functions main hardware components
1.5 Machine Types
1.6 Personal computer / workstation.
1.7 Mobile computer.
1.8 Smart device.
1.9 Super computer and server.
1.10 Application environment: Office automation (word processing, spreadsheet and presentation software); Programming; Management (stocks, accounting)

CHAPTER 2 (30 periods of which 10 are practical exercises)

General Computer Operation

2.1 The main units
- 2.1.1 Central unit
- 2.1.2 Central memory
- 2.1.3 Input and output units

2.2 What is a program?

2.3 Running a program

2.4 Number system and unit of data
- 2.4.1 Binary system
- 2.4.2 Convert binary to decimal and vice versa
- 2.4.3 Conversion from base 10 to other bases and vice versa
- 2.4.4 Octal-Binary and Binary-Octal conversion
- 2.4.5 Hexadecimal-Binary and Binary-Hexadecimal conversion
- 2.4.6 Data unit: Concept of bit; Byte and its multiples (Kilo, Mega, Giga, Tera, etc.)

2.5 Examples and exercises

CHAPTER 3 (8 periods of which 5 are practical exercises)

The processor

3.1 Presentation: brief overview of the key role of processors in modern computers

3.2 Processor Fundamentals
- 3.2.1 Definition of a processor and its role in executing instructions
- 3.2.2 Concepts of architecture and clock frequency

3.3 Processor Architecture
- 3.3.1 Details on the internal structure of a processor
- 3.3.2 Overview of pipelines and execution units

3.4 Advanced Processor Technologies
- 3.4.1 Discussion on technologies like hyper-threading, multi-core, and cache
- 3.4.2 Impact on performance and energy efficiency

3.5 Advanced Processor Features
- 3.5.1 Virtualization, encryption, and other advanced functionalities
- 3.5.2 Practical applications and benefits

3.6 Overclocking
- 3.6.1 Explanation of overclocking and its impact on performance
- 3.6.2 Risks and benefits, guidelines for safe overclocking

3.7 Benchmarking Exercises and Performance Analysis
- 3.7.1 Practical exercises to evaluate the performance of different processors
- 3.7.2 Use of benchmarking tools

CHAPTER 4 (7 periods of which 5 are practical exercises)

Memories

4.1 Introduction to PC Memory: brief overview of the importance of memory in computer systems

4.2 Types of PC Memory
- 4.2.1 Definition of RAM, ROM, and other types of memory
- 4.2.2 Role and differences between temporary and permanent memory

4.3 Memory Architecture
- 4.3.1 In-depth look at the internal structure of RAM and ROM
- 4.3.2 Overview of memory hierarchies

4.4 Functioning of Memory
- 4.4.1 Explanation of the read and write processes

4.4.2 Concepts of random access (RAM) and sequential access (ROM)
4.5 Advanced Types of Memory
4.5.1 Flash memory, cache, and other advanced technologies
4.5.2 Practical applications and benefits
4.6 Memory Management in Operating Systems
4.6.1 The role of the operating system in memory management
4.6.2 Concepts of paging, swapping, and virtual memory
4.7 Speed and Capacity Exercises for Memory
4.7.1 Practical exercises on optimizing memory performance
4.7.2 Use of memory diagnostic tools

CHAPTER 5 (12 periods of which 10 are practical exercises)

Motherboard and interfaces

5.1 Course Overview: importance of motherboards in computer systems
5.2 Role and Functions of Motherboards
5.2.1 Definition of the central role of motherboards in PC
5.2.2 Key functions and connections
5.3 Internal Structure of Motherboards
5.3.1 Physical structure of motherboards and functioning of various integrated components
5.4 Ports, Slots, and Connections
5.4.1 Description of common ports and slots
5.4.2 Connections and compatibility with other components
5.5 BIOS/UEFI
5.5.1 Role and functionality of boot firmware
5.5.2 Configuration and updating of BIOS/UEFI
5.6 Power Management
5.6.1 Concepts of energy saving and power management
5.6.2 Impact of settings on system stability
5.7 Integrated Video Cards
5.7.1 Role of integrated video cards in system graphics
5.7.2 Configuration and optimization of graphic performance
5.8 Integrated Audio Cards
5.8.1 Importance of integrated audio cards
5.8.2 Configuration and management of audio settings
5.9 External Connections
5.9.1 USB, HDMI, and other common external connections
5.9.2 Configuration and management of external peripherals
5.10 Storage Peripherals
5.10.1 Management of HDDs, SSDs, and optical drives
5.11 Integrated Networking
5.11.1 Role of integrated network cards
5.11.2 Configuration of network connections and troubleshooting
5.12 Wireless Technologies
5.12.1 Integrated Wi-Fi and Bluetooth
5.12.2 Configuration and management of wireless connections
5.13 Overclocking Motherboards
5.13.1 Explanation of overclocking and its risks/benefits
5.13.2 Safety techniques and performance optimization
5.14 Monitoring and Diagnostic Tools
5.14.1 Use of monitoring and diagnostic software
5.17 Motherboard Security

5.17.1 Protection against software and hardware threats
5.17.2 Security configuration and best practices
5.18 Thermal Management and Cooling
5.18.1 Thermal management techniques for motherboards
5.18.2 Choices and configuration of cooling systems
5.19 Exercises for motherboard
5.19.1 Practical exercises of configuring and optimizing a system based on a specific motherboard

CHAPTER 6 (10 periods of which 5 are practical exercises)

Mass memories

6.1 Course Overview: importance of mass storage devices in computing
6.2 Types of Mass Storage Devices
6.2.1 Definition of mass storage devices (HDDs, SSDs, flash memory etc.)
6.2.2 Role and differences between various types of storage
6.3 Internal Structure of Mass Storage Devices
6.3.1 Internal structure of HDDs and SSDs
6.4 Reading and Writing Data
6.4.1 Explanation of the read and write processes
6.4.2 Concepts of sequential and random access
6.5 RAID Configurations
6.5.1 Introduction to RAID (Redundant Array of Independent Disks)
6.5.2 Different RAID configurations and their benefits
6.6 Cloud Storage and Network Attached Storage (NAS)
6.6.1 Overview of cloud storage services
6.6.2 Understanding NAS and its applications
6.7 Storage Optimization
6.7.1 Practical exercises on optimizing storage performance
6.7.2 Use of diagnostic tools for storage analysis

CHAPTER 7 (18 periods of which 10 are practical exercises)

Input/Output peripherals

7.1 importance of computer peripherals in the user experience
7.2 Categories of Peripherals
7.2.1 Definition and classification of major categories of peripherals
7.2.2 Role of peripherals in a computer system
7.2.3 Introduction to Peripheral Drivers (Definition, The need for specific drivers for different peripherals, Procedure for Driver Installation, Updating and Maintenance of Drivers)
7.3 Common Input Devices
7.3.1 Keyboards, mouse, trackpads: role and functioning
7.3.2 Digitizers and Input Sensors: SCAN, Touchscreens (I/O), digital pens, and other input sensors
7.4 Common Output Devices
7.4.1 Monitors and Screens: types of monitors and available screens
7.4.2 Resolutions, refresh rates, and advanced technologies
7.4.3 Printers and Output Devices: different types of printers and output devices
7.5 Speakers and Headphones
7.5.1 Importance of audio in computer peripherals
7.5.2 Features and advanced configurations
7.6 Webcams and Multimedia Devices

7.6.1 Role of webcams and other multimedia peripherals
7.6.2 Use in video conferencing and multimedia production
7.7 Practical Peripherals Lab
7.7.1 Exercises on using various peripherals
7.7.2 Advanced configurations and problem-solving

CHAPTER 8 (30 periods of which 15 are practical exercises)

Operating systems installation

8.1 Introduction to Operating Systems
8.1.1. Definition of an operating system
8.1.2 Types of operating systems (Windows, Linux, macOS)
8.1.3 System requirements for installation
8.1.4 Partitioning and formatting concepts
8.2 Preparation for Installation
8.2.1 Creating installation media (USB, DVD)
8.2.2 Downloading Windows and Linux ISO images
8.2.3 Verifying ISO file integrity
8.3 Windows Installation
8.3.1 Booting from installation media
8.3.2 Installing Windows
8.3.3 Configuring installation options
8.3.4 Installing drivers and updates
8.3.5 Activating Windows
8.4 Linux Installation
8.4.1 Booting from Linux installation media
8.4.2 Installing a Linux distribution (e.g., Ubuntu)
8.4.3 Disk partitioning for Linux
8.4.4 Configuring GRUB (bootloader)
8.4.5 Installing drivers and updates
8.5 Post-Installation Configuration
8.5.1 Configuring networks and connections
8.5.2 Installing additional software
8.5.3 Configuring user accounts and passwords
8.5.4 Optimizing system settings
8.6 Troubleshooting and Q&A
8.6.1 Common installation issues and solutions
8.6.2 Diagnostic tools and problem-solving techniques
8.6.3 Student questions and answers
8.6.4 Tips for long-term operating system maintenance

APPLIED COMPUTING I
(120 PERIODS OF PRACTICAL EXERCISES)

OBJECTIVE

At the end of this subject, the student should be able to:

- Using word processing software
- Using spreadsheets
- Using presentations software

CHAPTER 1 (10 periods of practical exercises)

MS-Windows

1.1 Defining Files and Folders.
1.2 Creating Folders.
1.3 Copying Files & Folders Using Windows Explorer.
1.4 Moving Files & Folders Using Windows Explorer.
1.5 Transferring Files & Folders to Flash Drives or Other Storage Media.
1.6 Renaming and Deleting Files & Folders Using Windows Explorer.
1.7 Using Recycle Bin.
1.8 Displaying Files Description & Files Types.
1.9 Use of files in the cloud

CHAPTER 2 (10 periods of practical exercises)

LINUX

2.1 Defining Files and Folders Using the Terminal.
2.2 Creating Folders Using the Terminal.
2.3 Copying Files & Folders Using the Terminal.
2.4 Moving Files & Folders Using the Terminal.
2.5 Transferring Files & Folders to Flash Drives or Other Storage Media.
2.6 Renaming and Deleting Files & Folders Using the Terminal.
2.7 Displaying File Descriptions, File Types, and Permissions.

CHAPTER 3 (30 periods of practical exercises)

Word Processing Software

3.1 Exploring the Word Processor
- 3.1.1. Starting a Word processor.
- 3.1.2. Opening an existing file.
- 3.1.3. Moving a document.
- 3.1.4. Converting a document to an earlier version.
- 3.1.5. Viewing multiple pages.
- 3.1.6. Displaying thumbnails of pages.
- 3.1.7. Displaying or hiding non-printing characters.
- 3.1.8. Displaying a document in a different view.
- 3.1.9. Switching among open documents.
- 3.1.10. Viewing multiple open documents.
- 3.1.11. Opening a new document.
- 3.1.12. Saving a document.
- 3.1.13. Creating a new folder while saving a document.
- 3.1.14. Previewing how a document will look when printed.
- 3.1.15. Printing a document with default and custom settings.
- 3.1.16. Changing default program options.
- 3.1.17. Adding a button for a command to the Quick Access Toolbar.

3.2 Editing and Proofreading Documents
- 3.2.1 Selecting, deleting, copying, cutting, and pasting text.
- 3.2.2 Undoing an action, moving text by dragging.
- 3.2.3 Inserting the date and time.
- 3.2.4 Using the Thesaurus, Research, and Translate.
- 3.2.5 Displaying the document in different views.
- 3.2.6 Finding or replacing text.
- 3.2.7 Checking spelling and grammar.
- 3.2.8 Marking a document as final.

3.3 Changing the Look of Text
- 3.3.1 Previewing and applying styles.
- 3.3.2 Applying and copying character formatting.
- 3.3.3 Changing the font and font size.
- 3.3.4 Applying text effects, changing or highlighting text with color.
- 3.3.5 Inserting a line break.
- 3.3.6 Aligning paragraphs.
- 3.3.7 Indenting a paragraph or first line, increasing or decreasing indentation.
- 3.3.8 Adding a border or shading to a paragraph.
- 3.3.9 Formatting paragraphs as a list.
- 3.3.10 Changing style and the indent level of a list.
- 3.3.11 Sorting items in a list and creating a multilevel list.

3.4 Changing the Look of a Document
- 3.4.1 Adding a background color, background fill effects to a document.
- 3.4.2 Adding a text watermark and using a picture as a watermark.
- 3.4.3 Creating or saving a document as a template.
- 3.4.4 Inserting a header or footer in a document.
- 3.4.5 Inserting page numbers and changing the format of page numbers.
- 3.4.6 Inserting a page break, section break.
- 3.4.7 Adjusting page size, page margins, orientation.

3.5 Presenting Information in Columns and Tables
- 3.5.1 Formatting text in multiple columns, changing the width of columns.
- 3.5.2 Automatically columnizing text.
- 3.5.3 Inserting a column break.
- 3.5.4 Creating lists and sublists.
- 3.5.5 Inserting a table, merging cells, splitting cells, etc.
- 3.5.6 Adding, deleting cells, rows, or columns.
- 3.5.7 Converting text to a table or vice versa.
- 3.5.8 Inserting a quick table and applying a table style.
- 3.5.9 Inserting a spreadsheet worksheet.
- 3.5.10 Drawing a table.

3.6 Working with Graphics and Symbols
- 3.6.1 Inserting a picture, clip art, symbol, equation, etc.
- 3.6.2 Changing text wrapping, the position of a picture.
- 3.6.3 Adjusting the size of an object, moving, copying.
- 3.6.4 Inserting artistic text, drawing a shape, etc.

3.7 Working with Diagrams and Charts
- 3.7.1 Inserting a diagram.
- 3.7.2 Adding text and shape to a diagram.
- 3.7.3 Resizing, moving, and changing the layout and style.
- 3.7.4 Inserting a chart.
- 3.7.5 Entering data in a new chart, reading a chart.
- 3.7.6 Editing data, changing chart type, changing chart style.

3.7.7 Fitting a column width, gridlines, etc.

3.8 Working with Longer Documents

3.8.1 Adding, deleting, inserting a building block.
3.8.2 Creating and updating a table of contents.
3.8.3 Creating and updating a table of figures.
3.8.4 Marking an index entry and creating an index.
3.8.5 Inserting a hyperlink, editing, etc.
3.8.6 Creating a bibliography.
3.8.7 Creating a footnote or endnote.

3.9 Creating Form Letters, Email, Messages, and Labels

3.9.1 Sorting data, filtering records in a data source.
3.9.2 Printing an envelope based on an address in a document.
3.9.3 Sending personalized email messages.
3.9.4 Creating mailing labels.

3.10 Collaborating with Others

3.10.1 Sending a copy of a document as an email attachment.
3.10.2 Using track changes.
3.10.3 Accepting or rejecting a change in a document.
3.10.4 Opening and closing the reviewing pane.
3.10.5 Inserting, editing, deleting, hiding, or responding to a comment.
3.10.6 Combining versions of a document.
3.10.7 Hiding a reviewer's changes.
3.10.8 Accepting all changes in a document.
3.10.9 Protecting a document with a password.
3.10.10 Removing a password.
3.10.11 Restricting formatting and editing.

CHAPTER 4 (50 periods of practical exercises)

Spreadsheets Software

4.1 Setting up a Workbook

4.1.1 Opening a Workbook.
4.1.2 Creating a new Workbook.
4.1.3 Saving a Workbook.
4.1.4 Setting file properties.
4.1.5 Defining custom properties.
4.1.6 Defining a worksheet.
4.1.7 Creating a new worksheet.
4.1.8 Renaming a worksheet.
4.1.9 Copying a worksheet to another workbook.
4.1.10 Changing the order of worksheets in a workbook.
4.1.11 Hiding, unhiding, deleting a worksheet.
4.1.12 Changing a row's height or column width.
4.1.13 Inserting, deleting, hiding, or unhiding a column or row.
4.1.14 Inserting or deleting a cell.
4.1.15 Moving a group of cells to a new location.
4.1.16 Zooming in or out on a worksheet.
4.1.17 Zooming in or out to a specific zoom level.
4.1.18 Changing to another open workbook.
4.1.19 Arranging all open workbooks in the program window.
4.1.20 Adding, moving, or removing a button to the Quick Access Toolbar.

4.2 Working with Data and Data Tables

4.2.1 Entering a data series using Autofill.

4.2.2 Changing how dragging the fill handle extends a series.
4.2.3 Copying and pasting cells.
4.2.4 Copying and pasting a row or column.
4.2.5 Finding data within a worksheet.
4.2.6 Replacing a value with another value within a worksheet.
4.2.7 Editing a cell's contents by hand.
4.2.8 Checking spelling.
4.2.9 Looking up a word in the Thesaurus.
4.2.10 Translating a word to another language.
4.2.11 Creating a data table.
4.2.12 Adding rows to a data table.
4.2.13 Resizing a table.
4.2.14 Adding a Total row to a column.
4.2.15 Changing the total row summary function.
4.2.16 Renaming a table.

4.3 Performing Calculations on Data

4.3.1 Creating a named range.
4.3.2 Creating a named range from a selection.
4.3.3 Displaying the name manager.
4.3.4 Editing a named range.
4.3.5 Creating a formula.
4.3.6 Creating a formula using the Insert function dialog box.
4.3.7 Using a named range in a formula.
4.3.8 Referring to a table column or row in a formula.
4.3.9 Creating a formula using Formula AutoComplete.
4.3.10 Creating a formula that doesn't change when copied between cells.
4.3.11 Creating a formula that does change when copied between cells.
4.3.12 Creating a conditional formula.
4.3.13 Displaying cells that provide values for a formula.
4.3.14 Displaying formulas that use a cell's contents.
4.3.15 Removing tracer arrows.
4.3.16 Locating errors in a worksheet.
4.3.17 Stepping through a formula to locate an error.

4.4 Changing Document Appearance

4.4.1 Changing a cell's font, font style, font color, or background color.
4.4.2 Adding a border to a cell.
4.4.3 Applying a style to a cell.
4.4.4 Creating or deleting a new style.
4.4.5 Copying a cell's formatting onto another cell.
4.4.6 Applying a workbook theme.
4.4.7 Changing theme fonts, colors, and graphic effects.
4.4.8 Saving a workbook's format as a new theme.
4.4.9 Creating a new table style.
4.4.10 Formatting a cell value as a phone number.
4.4.11 Formatting cell data as a currency value.
4.4.12 Selecting a foreign currency symbol.
4.4.13 Adding words to a cell's value.
4.4.14 Applying a conditional format to a cell.
4.4.15 Editing or deleting a conditional formatting rule.
4.4.16 Displaying data bars in one or more cells.
4.4.17 Displaying a color scale in one or more cells.
4.4.18 Displaying icon sets in one or more cells.

4.4.19 Adding a picture to a worksheet.
4.4.20 Changing a picture's characteristics.

4.5 Focusing on Specific Data Using Filters
4.5.1 Applying a filter to a worksheet.
4.5.2 Clearing a filter.
4.5.3 Displaying the top or bottom values in a column.
4.5.4 Creating a custom filter.
4.5.5 Generating a random value.
4.5.6 Generating a random value between two other values.
4.5.7 Summarizing data quickly using AutoCalculate.
4.5.8 Summarizing filtered data using a SubTotal formula.
4.5.9 Finding list rows that contain unique values.
4.5.10 Creating a validation rule.
4.5.11 Identifying which cells contain invalid data.
4.5.12 Turning off data validation in a cell.

4.6 Reordering and Summarizing Data
4.6.1 Sorting a data list.
4.6.2 Sorting a data list by value in multiple columns.
4.6.3 Adding or deleting a sorting level.
4.6.4 Creating a custom list for sorting.
4.6.5 Sorting worksheet data by a custom list of values.
4.6.6 Organizing worksheet data into groups.
4.6.7 Showing and hiding levels of detail in a grouped data list.
4.6.8 Removing grouping levels from a data list.
4.6.9 Looking up data in a data list.

4.7 Combining Data from Multiple Sources
4.7.1 Creating a workbook template.
4.7.2 Creating a new workbook that is based on a template.
4.7.3 Creating a worksheet template.
4.7.4 Adding a template-based worksheet to a workbook.
4.7.5 Creating a link between two cells.
4.7.6 Opening multiple workbooks simultaneously.

4.8 Create Charts and Graphics
4.8.1 Creating a chart.
4.8.2 Changing how spreadsheet plots your data.
4.8.3 Removing a series from an axis.
4.8.4 Adding a series to an axis.
4.8.5 Moving a chart to its own worksheet.
4.8.6 Applying a Chart Style to a chart.
4.8.7 Applying a different layout to a chart.
4.8.8 Changing the appearance of a chart's gridlines.
4.8.9 Selecting a chart element for formatting.
4.8.10 Selecting a data point in a series.
4.8.11 Formatting a chart element.
4.8.12 Saving a chart as a chart template.
4.8.13 Creating a SmartArt diagram.
4.8.14 Adding text to a diagram shape.
4.8.15 Adding a shape to a diagram.
4.8.16 Changing the format of a diagram shape.

4.9 Printing
4.9.1 Displaying a worksheet in Page Layout View.
4.9.2 Adding a header or footer to a worksheet.

4.9.3 Creating an Auto Header.
4.9.4 Adding an image to a header or footer.
4.9.5 Formatting an image in a header or footer.
4.9.6 Changing a worksheet's margins.
4.9.7 Changing a worksheet's page orientation.
4.9.8 Printing a worksheet on a specific number of pages.
4.9.9 Previewing a worksheet before printing.
4.9.10 Adding a page break to a worksheet.
4.9.11 Removing a page break from a worksheet.
4.9.12 Changing the order in which worksheets print.
4.9.13 Printing a worksheet or part of a worksheet.

4.10 Working with Other Microsoft Office System Programs
4.10.1 Linking to another Microsoft Office System document.
4.10.2 Embedding another document in a workbook.
4.10.3 Viewing a linked or embedded document.
4.10.4 Creating, editing, or deleting a hyperlink.

4.11 Collaborating with Colleagues
4.11.1 Turning on workbook sharing.
4.11.2 Adding a comment to a cell.
4.11.3 Editing and deleting a comment.
4.11.4 Tracking changes made to a workbook.
4.11.5 Accepting and rejecting changes.
4.11.6 Recording workbook changes on a History worksheet.
4.11.7 Requiring a password to open a workbook.
4.11.8 Publishing a workbook to the web.

CHAPTER 5 (20 periods of practical exercises)

Presentation Software

5.1 Creating a presentation
5.1.1 Starting and exploring the program
5.1.2 Moving around a presentation
5.1.3 Previewing slides in slide sorter view
5.1.4 Reversing an action
5.1.5 Saving a presentation

5.2 Working with a presentation
5.2.1 Creating a new presentation using a design template
5.2.2 Creating a new slide
5.2.3 Rearranging slides in slide sorter view
5.2.4 Viewing a slideshow

5.3 Adding and modifying text
5.3.1 Selecting and deselecting objects
5.3.2 Formatting text
5.3.3 Adding text to slides
5.3.4 Adjusting position of text objects

5.4 Using a color scheme
5.4.1 Adding a background

5.5 Inserting information into PowerPoint
5.5.1 Inserting a clip art image
5.5.2 Inserting and modifying a picture

5.6 Producing a slideshow
5.6.1 Setting slide transitions

5.6.2 Animating slide text
5.7 Creating a multimedia presentation
5.7.1 Inserting sounds in a presentation
5.7.2 Inserting videos in a presentation
5.8 Printing a presentation
5.8.1 Printing slides for note taking
5.9 Applying a modifying templates
5.9.1 Understanding and applying templates
5.9.2 Using PowerPoint masters
5.10 Using a color scheme
5.10.1 Adding a background
5.10.2 Creating a textured background
5.11 Drawing an modifying objects
5.11.1 Drawing an object
5.11.2 Editing an object
5.11.3 Copying and moving an object
5.11.4 Changing the shape of an object
5.12 Inserting information into PowerPoint
5.12.1 Inserting a clip art image
5.12.2 Inserting and modifying a picture
5.12.3 Inserting and modifying word art
5.13 Producing a slideshow
5.13.1 Setting slide transitions
5.13.2 Animating slide text
5.13.3 custom animation
5.14 Creating a multimedia presentation
5.14.1 Inserting sounds in a presentation
5.14.2 Setting slide timings

INTRODUCTION TO ELECTRICITY AND ELECTRONICS
(30 PERIODS OF PRACTICAL EXERCISES)

OBJECTIVE

At the end of this course the student should be able to:

- Understand how to protect individuals against electric shocks.
- Use measuring devices effectively.
- Comprehend resistance and their measurement techniques.
- Identify different types of capacitors.
- Understand the roles and operation of diodes, LED diodes, and transformers.

CHAPTER 1 (4 periods of practical exercises)

Electrical Safety

1.1 Intensity is the essential cause of danger.
1.2 The circumstances of electrocution.
  1.2.1 By direct contact.
  1.2.2 By indirect contact.
  1.2.3 Protection of persons against electric shocks.
  1.2.4 Scenarios involving direct contact.
  1.2.5 Scenarios involving indirect contact.

CHAPTER2 (4 periods of practical exercises)

Soldering of electronic components on a perforated plate

2.1 Materials (soldering iron, perforated plate, tin).
2.2 Soldering of electronic components on perforated plates.

CHAPTER3 (3 periods of practical exercises)

Use of measuring devices

3.1 General information on measuring devices with multimeters.
3.2 Measurement and adjustment of direct and alternating voltages with a digital voltmeter.
3.3 Measurement and adjustment of direct and alternating currents with a digital ammeter.
3.4 Continuity test (conductor, fuse, etc.).

CHAPTER4 (4 periods of practical exercises)

Electrical resistances

4.1 Reading the resistor color code.
4.2 Direct resistance measurement.
4.3 Comparison between measured values and read values.
4.4 Grouping of Resistors: Series, Parallel, and Mixed Configurations.
4.5 Measurement of currents and voltages in the three types of groupings above.

CHAPTER5 (3 periods of practical exercises)

Capacitor

5.1 Types of capacitors (polarized and non-polarized) and their units.
5.2 Charging and discharging a capacitor (RC circuit).

CHAPTER6 (3 periods of practical exercises)

The oscilloscope

6.1 Observe the alternating signal on the oscilloscope, measure the period, the frequency, Peak to peak value, effective value, average and maximum value.

CHAPTER7 (3 periods of practical exercises)

Diode and Diode LED

7.1 Forward and Reverse Biasing of a Diode.
7.2 Testing a diode and/or a diode led.

CHAPTER 8 (2 periods of practical exercises)

Transformer

8.1 Definition, role, constitution and symbol
8.2 Operating principle
8.3 Transformation Ratio.

CHAPTER 9 (4 periods of practical exercises)

Power supply

9.1 Block diagram of a power supply.
9.2 Role of each block.
9.3 Shape of the signals at the output of each block.

SMART MOBILE DEVICE I
(60 PERIODS OF PRACTICAL EXERCISES)

CHAPTER 1 (2 periods of practical exercises)

Overview of Smart Mobile Devices

1.1 Smartphones and Tablets Overview.
1.2 Features and Capabilities

CHAPTER 2 (2 periods of practical exercises)

Possible Connections with a Smartphone.

2.1 Wi-Fi Connection.
2.2 Hotspot Connection Sharing.
2.3 Bluetooth Connection Sharing.
2.4 USB Connection sharing.
2.5 3G, 4G and 5G Connections.
2.6 USB OTG (On-The-Go) Cable Connection.
2.7 NFC Connection. (Near Field Communication) Short range Wi-Fi.
2.8 Direct Wi-Fi Connection.
2.9 Drone Control, Lighting Control, and Home Automation.
2.10 Infrared Port for transferring files and emulating Remote TV.
2.11 HDMI Connection.
2.12 GPS For Sharing Locations.

CHAPTER 3 (2 periods of practical exercises)

3.1 Showing Diagrams Of Smartphone Components

CHAPTER 4 (5 periods of practical exercises)

Network Concepts

4.1 Cellular Data GSM, 3G, 4G, and 5G.
4.2 Wi-Fi Network.
4.3 Bluetooth Network.
4.4 NFC Network.

CHAPTER 5 (2 periods of practical exercises)

Operating Systems

5.1 IOS, Android, and Windows Overview.
5.2 Basic settings.

CHAPTER 6 (4 periods of practical exercises)

Device Security

6.1 Password (Complexity, Length, and Multi-Factor Authentication).
6.2 Biometrics (Fingerprint, Face Recognition, and Iris Scan).
6.3 Encryption (Data, Messages, and End to end encryption).
6.4 Mobile Device Management Security for Enterprise.

CHAPTER 7 (6 periods of practical exercises)

Mobile Configuration (IOS and Android)

7.1 Select Language.
7.2 Connect to the internet.
7.3 Log in with Apple / Google account (Create one).
7.4 Scheduling Basic Features and Settings (Location, Date and Time, etc...).
7.5 Backup and Restore (Cloud Storage Overview).
7.6 Mail Synchronization (Cloud Workspace).

CHAPTER 8 (6 periods of practical exercises)

Configuring Network Settings: (IOS and Android)

8.1 Wi-Fi Settings.
8.2 Mobile Data Settings.
8.3 Bluetooth Settings.
8.4 VPN Settings.
8.5 HOTSPOT Settings.
8.6 Location Services.
8.7 Reset Network Settings.

CHAPTER 9 (4 periods of practical exercises)

Emails in: (IOS and Android)

9.1 Configure more than one Email (Outlook, Gmail, Yahoo, etc...)
9.2 Data Synchronization.
9.3 Security measures for Emails.

CHAPTER 10 (4 periods of practical exercises)

Smartphone usage precautions

4.8 Privacy Settings.
4.9 App Permissions.
4.10 Backup Data.
4.11 Install Anti Malware.
4.12 Location Services.
4.13 Parental Control (Children protection).
4.14 Emergency Contacts.

CHAPTER 11 (1 period of practical exercises)

Tips to Save Battery Life

11.1 Screen Settings.
11.2 Turn Off Background Apps.
11.3 Avoid High Temperature.
11.4 Disable Location Services.
11.5 Monitor Battery Health.

CHAPTER 12 (2 periods of practical exercises)

Backup & Reset

12.1 Backup My Data.
12.2 Backup Account.
12.3 Automatic Restore.
12.4 Factory Data Reset.

CHAPTER 13 (4 periods of practical exercises)

Integration Of Cloud Solutions

13.1 Google Photos.
13.2 DropBox.
13.3 OneDrive.
13.4 Other Solutions.

CHAPTER 14 (10 periods of practical exercises)

Enterprise Mobility Overview

14.1 Importance of enterprise mobility:
- 14.1.1 Increasing productivity.
- 14.1.2 Flexibility.
- 14.1.3 Communication Performance.

14.2 Overview and concepts:
- 14.2.1 Mobile Device Management (MDM).
- 14.2.2 Mobile Application Management (MAM).
- 14.2.3 Bring Your Own Device (BYOD), and others.
- 14.2.4 Enterprise App Development.
- 14.2.5 Mobile Content Management (MCM).
- 14.2.6 Integration of mobile solutions with enterprise systems.
- 14.2.7 Practical Exercises (Creating a simulated MDM environment).

CHAPTER 15 (6 periods of practical exercises)

Mobile Security Overview of an Enterprise Environment

15.1 Crucial aspects of Mobile Security in an Enterprise Environment.
15.2 Importance of latest updates on smartphones (Security Patches).
15.3 Concepts of Data encryption.
15.4 Discussion about security and communication for enterprise.
15.5 Development and aligning Security policies for enterprise.
15.6 Demonstrate settings configuration for maximum security.
15.7 Discussions and questions about a case study.

COMPUTER ASSEMBLY I
(30 PERIODS OF PRACTICAL EXERCISES)

CHAPTER 1 (2 periods of practical exercises)

Computer Components Summary

Introduction to hardware components and their functions.

1.1 Introduction:

Outline the session's objectives.

1.2 Importance and Summary:

Present and discuss their roles and importance:

- Central Processing Unit (CPU)
- Motherboard
- Random Access Memory (RAM)
- Storage Devices (HDD/SSD)
- Power Supply Unit (PSU)
- Graphics Processing Unit (GPU)
- Peripheral Devices
- Cooling Systems
- Connectors and Ports
- BIOS/UEFI Firmware

CHAPTER 2 (1 period of practical exercises)

Safety Measures

Proper handling of components, anti-static precautions.

2.1 Importance:

2.1.1 Safety to prevent damage of computer components.

2.1.2 Discuss possible risks associated with electrostatic discharge (ESD).

2.2 Anti-Static Precautions Summary:

2.2.1 What electrostatic discharge (ESD) is and why it can damage computer components.

2.2.2 Anti-static measures, and usage of anti-static wrist straps.

2.3 Demonstration of Anti-Static Wrist Straps:

2.3.1 Properly wearing an anti-static wrist strap.

2.3.2 Correct connection to a grounded surface.

2.3.3 Needs of the strap during the assembly process.

2.3.4 Distribute anti-static wrist straps to each student.

2.3.5 Wearing straps and connecting to a grounded surface.

2.3.6 Correct way to connect and disconnect components while wearing it.

2.3.7 Discussion for handling components.

CHAPTER 3 (1 period of practical exercises)

Equipment and Tools

Identification and usage of tools.

3.1 Introduction:
- 3.1.1 Importance of using the right tools.
- 3.1.2 How using tools to ensure efficiency.

3.2 Summary and demonstration of Main Tools:
- 3.2.1 Present a list of main tools required. (screwdrivers, wrist straps, cable cutters, crumping tool, testers…).
- 3.2.2 The function of each tool.
- 3.2.3 Usage of each tool.
- 3.2.4 Demonstration of using a screwdriver for different types of screws, etc.
- 3.2.5 Proper handling to avoid injuries.
- 3.2.6 Distribute tools to each student or group.
- 3.2.7 Practice using each tool on some components.

3.3 Safety Discussion:
- 3.3.1 Discuss safety measures when using tools.
- 3.3.2 Importance of arranging tools in good condition.

CHAPTER 4 (3 periods of practical exercises)

Component Identification

Activity: Identify and handle components.

4.1 Introduction:
- 4.1.1 Key computer components covered in first session.
- 4.1.2 Importance of hands-on experience.

4.2 Workstation Setup:
- 4.2.1 Equipe Workstation with necessary components.
- 4.2.2 Check if each workstation has a grounded surface and anti-static wrist straps.

4.3 Demonstration, Guidelines and Identification:
- 4.3.1 Good handling of a component.
- 4.3.2 Wear anti-static wrist straps.
- 4.3.3 Divide students into small groups, one to a workstation.
- 4.3.4 Distribute components and labels to each group.
- 4.3.5 Identify and label each component.
- 4.3.6 Rotate groups through different workstations to check various components.

4.4 Advanced Challenges:
- 4.4.1 Advanced Information (e.g., different CPU sockets, RAM types).

4.5 Group Discussion:
- 4.5.1 Small groups students to discuss problems and share solutions.

4.6 Review:
- 4.6.1 Correct identification of each component.

CHAPTER 5 (1 periods of practical exercises)

Basic Assembly

5.1 Setup Workspace
- 5.1.1 Creating a clean and organized workspace.

5.2 Workspace Fundamentals:
- 5.1.2 Importance of an organized workspace in assembly process.
- 5.1.3 Efficiency, safety, and the prevention of component damage.
- 5.1.4 List of items for an assembly workspace (anti-static mats, lighting, tools, etc.).
- 5.1.5 Importance of cable management, tool placement, and a clear work surface.

5.3 Workspace Setup Demonstration:
- 5.1.6 Process of setting up an organized workspace.
- 5.1.7 Tips on cable management and tools.
- 5.1.8 Students Organize their workstations.
- 5.1.9 Discuss and share ideas for an efficient setup.

5.4 Cable Management Techniques and Safety:
- 5.1.10 Discuss cable management techniques to keep the workspace clean.
- 5.1.11 Examples of cable organizers, ties, and clips.
- 5.1.12 Review safety considerations in the workspace.
- 5.1.13 Groups visit other's workspaces.

CHAPTER 6 (3 periods of practical exercises)

Power Supply, Motherboard and Processor Installation

Installing PSU, Motherboard and CPU.

6.1 Introduction:
- 6.1.1 Importance of PSU, Motherboards and CPU in a system.
- 6.1.2 Importance of following steps during installation.

6.2 Power Supply Unit Installation:
- 6.2.1 Function and role of PSU.
- 6.2.2 Steps for installing a PSU.
- 6.2.3 Distribute power supply units to each student or group.
- 6.2.4 Students to install the PSU following steps.
- 6.2.5 Discuss problems.

6.3 Motherboard Installation:
- 6.3.1 Role and function of the motherboard and all its components.
- 6.3.2 Demonstration of the process while installing it.
- 6.3.3 Importance of using standoffs to prevent short circuits.
- 6.3.4 Distribute motherboards to each student or group.
- 6.3.5 Students should install the motherboard in their workstations.
- 6.3.6 Questions with discussion.

6.4 Central Processing Unit Installation:
- 6.4.1 Role and types of CPU.
- 6.4.2 Demonstrate the steps for installing a CPU and the cooling system.
- 6.4.3 Check its compatibility with Motherboards.
- 6.4.4 Distribute CPU to each student or group.
- 6.4.5 Students should install the CPU following steps.
- 6.4.6 Discuss problems.

6.5 Cable Management inside workstation:
- 6.5.1 Cable management techniques with examples.

CHAPTER 7 (3 periods of practical exercises)

Memory, Storage Device, VGA and Network Installation:

Installing Memory, Storage Device, VGA and Network.

7.1 Introduction:
- 7.1.1 Importance of RAM, Storage Device, VGA and Network.
- 7.1.2 Handling components with care.

7.2 RAM Installation:
- 7.2.1 Role of RAM.
- 7.2.2 Demonstrate of handling RAM in proper way.
- 7.2.3 Types of RAM (e.g., DDR3, DDR4) and their compatibility with different motherboards.
- 7.2.4 Distribute RAM to each student or group.
- 7.2.5 Install RAM.
- 7.2.6 Discussion and questions related to RAM installation.

7.3 Storage Installation:
- 7.3.1 Purposes and types of storage devices (HDD, SSD…).
- 7.3.2 Importance of aligning and securing storage devices properly.
- 7.3.3 Demonstrate the process of installing storage devices.
- 7.3.4 Show cable management and explain storage capacity.
- 7.3.5 Distribute storage devices to each student or group.
- 7.3.6 Installing storage devices.
- 7.3.7 Discussion and questions related to storage device installation.

7.4 VGA Installation:
- 7.4.1 Role of VGA.
- 7.4.2 Different types of VGAs (Dedicated or Shared “built in”) and their compatibility.
- 7.4.3 Distribute VGA to each student or group.
- 7.4.4 Installation of VGA.
- 7.4.5 Discussion and questions related to VGA installation.

7.5 Network Installation:
- 7.5.1 Role of Network.
- 7.5.2 Different types of Networks (separated card or built in) and their compatibility.
- 7.5.3 Distribute Network card to each student or group.
- 7.5.4 Installation of a Network.
- 7.5.5 Discussion and questions related to Network installation.

CHAPTER 8 (3 periods of practical exercises)

Peripheral Devices

Connecting peripherals and basic configuration.

8.1 Introduction
- 8.1.1 Role of peripheral devices.
- 8.1.2 Types and purposes of peripherals.

8.2 Summary and Demonstration:
- 8.2.1 Common peripheral devices such as keyboards, mice, monitors, printers, etc...
- 8.2.2 Different types of connectors (USB, HDMI, etc.) used for peripherals.
- 8.2.3 Process of connecting peripherals.
- 8.2.4 Importance of connecting devices to the correct ports.
- 8.2.5 Distribute peripheral devices to each student or group.
- 8.2.6 Instruction to connect peripherals.

8.3 Basic Configuration Overview:

8.3.1 Basic configurations for peripheral devices.
8.3.2 Students should adjust peripheral components.
8.3.3 Discussion and questions related to connecting and configuring peripherals.

8.4 Peripheral Troubleshooting Tips:
8.4.1 Discuss problems that may arise during peripheral connection and configuration.
8.4.2 Provide troubleshooting tips to solve problems.
8.4.3 Assign groups to check other peripheral connections and configurations.
8.4.4 Importance of testing peripherals for proper functionality.

CHAPTER 9 (1 periods of practical exercises)

Cable Management

Proper arrangement of cables.

9.1 Benefits of Cable Management:
9.1.1 Importance of cable management.
9.1.2 Advantages of proper cable management.
9.1.3 Discuss issues that arise from poor cable management.

9.2 Basic Cable Types and Connectors:
9.2.1 Common cable types and connectors.
9.2.2 Purpose of each cable and its usage.

9.3 Demonstration of Cable Management Principles and techniques:
9.3.1 Cable management principles.
9.3.2 Examples of well-managed cables.
9.3.3 Distribute cable organizers, ties, and clips to each student or group.
9.3.4 Students should organize, arrange and route cables.
9.3.5 Discuss issues related to cables.
9.3.6 Discussion of advanced cable management techniques, such as custom cable lengths, cable sleeving, and using cable combs.

9.4 Group Assessment and Feedback:
9.4.1 Assign groups to evaluate each other's cable management.
9.4.2 Summarize the techniques of proper cable management.
9.4.3 Emphasize the impact of organized cables on system performance and maintenance.

CHAPTER 10 (6 periods of practical exercises)

Software Installation

Operating System Installation (Windows and Linux)

Installing an OS and basic driver setup.

10.1 Introduction:
10.1.1 Importance of the OS in functioning the computer.
10.1.2 Role of drivers in connecting hardware components to the operating system.

10.2 Selection of Operating System and Installation Process:
10.2.1 Compare different operating systems (Windows and Linux).
10.2.2 Preparing the installation media for the OS.
10.2.3 Overview of the steps involved in installing an operating system.
10.2.4 Explanation of partitioning, formatting, and selecting installation options.

10.3 Demonstration of OS Installation:
10.3.1 Demonstrate the process of installing OS.
10.3.2 Partitioning, formatting, and making essential configuration choices.
10.3.3 Distribute installation media to each student or group.

10.3.4 Installation of OS.

10.4 Driver Setup and Installation:

10.4.1 Concept of drivers and their role.

10.4.2 Importance of installing main drivers.

10.4.3 Demonstrate the process of installing basic drivers.

10.4.4 Find drivers and best practices for installation.

10.5 Hands-On Driver Setup:

10.5.1 Distribute driver installation discs to each student or group.

10.5.2 Install basic drivers.

10.5.3 Assign groups to assess each other's.

CHAPTER 11 (2 periods of practical exercises)

BIOS/UEFI Configuration

Accessing and configuring basic BIOS/UEFI settings.

11.1 Introduction:

11.1.1 Role of the BIOS/UEFI in managing system.

11.1.2 Importance of configuring these settings.

11.2 Difference Between BIOS and UEFI:

11.2.1 Differences between BIOS (Basic Input/Output System) and UEFI (Unified Extensible Firmware Interface).

11.2.2 Advantages of UEFI, such as support for larger storage devices.

11.3 Accessing BIOS/UEFI Settings:

11.3.1 Demonstrate how to access the BIOS/UEFI on different systems.

11.3.2 Common settings found in the BIOS/UEFI.

11.3.3 Demonstrate the process of configuring settings in the BIOS/UEFI.

11.3.4 Students should access the BIOS/UEFI.

11.3.5 Navigate the BIOS/UEFI interface.

11.3.6 Questions and discussion impact of these settings on system.

CHAPTER 12 (3 periods of practical exercises)

Testing and Troubleshooting

Powering up the system, diagnostic tests, and troubleshooting.

12.1 Introduction:

12.1.1 Importance of system testing.

12.1.2 Purpose of troubleshooting.

12.2 Power-Up and Diagnostic Testing for the system:

12.2.1 Demonstrate the proper procedure for powering up the system.

12.2.2 Importance of checking all signs of life.

12.2.3 Introduce diagnostic tests.

12.2.4 Importance of checking error messages.

12.2.5 Students power up their computer systems.

12.2.6 Performing diagnostic tests.

12.2.7 Discussions and questions related to system testing.

12.3 Introduction, demonstration and discussion of Troubleshooting:

12.3.1 Importance of troubleshooting.

12.3.2 Systematic process to troubleshooting.

12.3.3 Demonstrate the process of troubleshooting.

12.3.4 Error messages and diagnostic tools.

12.3.5 Students should experience and simulate common issues in their systems for troubleshooting practice.

12.3.6 Resolving the issues.
12.3.7 Discussion the process and steps taken during troubleshooting.

CHAPTER 13 (1 period of practical exercises)

Finalizing the Build

Updating software and optimizing system performance.

13.1 Introduction:
- 13.1.1 Importance of updating OS and general updates.

13.2 Operating System Updates:
- 13.2.1 Importance of keeping the operating system up-to-date.
- 13.2.2 Checking for updates and install the latest.

13.3 Driver Updates:
- 13.3.1 Importance of updating hardware drivers for compatibility.
- 13.3.2 Demonstrate how to check for and install the latest drivers.

13.4 Additional Software Installation:
- 13.4.1 Process of installing additional software.
- 13.4.2 Students installation of main applications.
- 13.4.3 Ensure proper installation.

13.5 System Tips and discussion:
- 13.5.1 Tips for optimizing system performance.
- 13.5.2 Importance of regular maintenance tasks, (disk cleanup….).
- 13.5.3 Implementing system improvement tips.
- 13.5.4 Discussion and questions related to the finalization process.
- 13.5.5 Emphasize the importance of ongoing system maintenance and updates.

PROGRAMMING I
(90 PERIODS OF WHICH 60 ARE PRACTICAL EXERCISES)

This curriculum balances theoretical knowledge with practical application, ensuring students understand the fundamentals of Python and programming.
Teacher's notes:

- Continuous assessment through quizzes, assignments, and practical exercises.
- End-of-module small projects to reinforce learning and practical application.
- Guest Lectures/Workshops with experienced programmers or industry professionals (if possible / available).
- Peer Programming: Encourage collaborative learning and peer-to-peer teaching.
- Online Resources: Online resources, tutorials, and forums for additional learning.

CHAPTER 1 (5 periods)
Introduction to Python and Computer Programming

1.1 What is Programming?
1.2 Why Python?
1.3 Python Installation and Setup
1.4 Basic Syntax and Hello World
1.5 Introduction to IDEs and Text Editors
1.6 Writing and Running Python Scripts

CHAPTER 2 (10 periods of which 5 are practical exercises)
Data Types, Variables, Basic I/O, and Basic Operators

2.1 Understanding Data Types: Integers, Floats, Strings, Booleans
2.2 Variables and Naming Conventions
2.3 Basic Input and Output Operations
2.4 Arithmetic Operators and Their Use
2.5 Order of Execution and Parentheses in Arithmetic Operations
2.6 Introduction to Python's Built-in Functions (including print, input)

CHAPTER 3 (15 periods of which 10 are practical exercises)
Boolean Values, Conditional Execution, Loops, Lists, and Logical Operations

3.1 Boolean Logic (True, False)
3.2 Conditional Statements (if, elif, else)
3.3 Loop Structures: for and while Loops
3.4 Introduction to Lists: Creation, Access, and Manipulation
3.5 List Comprehensions
3.6 Logical and Bitwise Operators

CHAPTER 4 (15 periods of which 10 are practical exercises)

Functions, Tuples, Dictionaries, and Data Processing

4.1 Defining and Calling Functions
4.2 Function Arguments and Return Values
4.3 Introduction to Tuples and Dictionaries
4.4 Data Processing Techniques
4.5 Lambda Functions and Map/Filter/Reduce

CHAPTER 5 (15 periods of which 10 are practical exercises)

Modules, Packages, and PIP

5.1 Understanding Python Modules
5.2 Creating and Using Custom Modules
5.3 Introduction to Python Packages
5.4 Using PIP for Installing and Managing Packages
5.5 Virtual Environments in Python

CHAPTER 6 (15 periods of which 10 are practical exercises)

Strings, String and List Methods, Exceptions

6.1 String Manipulation and Methods
6.2 List Methods and Operations
6.3 Introduction to Exception Handling
6.4 Try, Except, Finally Blocks
6.5 Raising Exceptions

CHAPTER 7 (15 periods of practical exercises)

Project Work

8.1 A final project encompassing all learned aspects to be completed individually or in groups.

PROGRAMMING THE WEB I
(30 PERIODS OF PRACTICAL EXERCISES)

This course focuses on giving a thorough understanding of HTML, preparing students for future courses on CSS and Javascript.
Teaching Instructions:

- Each session should include practical exercises.
- Encourage teamwork and peer learning wherever possible.
- The recommended tool for web client programming is currently Visual Studio Code.
- Regular tests and assignments should be performed, to track student progress.

CHAPTER 1 (12 periods of practical exercises)

HTML Basics

1.1 Understanding Web Pages
- 1.1.1 Basic concept of the Internet, websites, and web pages
- 1.1.2 Web site hosting and browsers
- 1.1.3 Introduction to HTML and its significance in web development

1.2 Basic syntax of HTML
- 1.2.1 HTML elements and tags, the Document Object Model, attributes, opening and closing tags
- 1.2.2 Structure of a simple HTML page (doctype, html, head, body)
- 1.2.3 Choosing the right charset
- 1.2.4 Brief history of HTML

1.3 Basic HTML Elements
- 1.3.1 Paragraphs (div, p, span), headings, and text formatting
- 1.3.2 Emphasis, strong, and basic text styles

1.4 Creating Lists
- 1.4.1 Ordered and unordered lists
- 1.4.2 Nested lists and list attributes

1.5 Adding Images and Inserting Links
- 1.5.1 Adding images to a webpage (img)
- 1.5.2 Image attributes (src, alt, width, height)
- 1.5.3 Using anchor (a) tags to create links
- 1.5.4 Absolute vs. relative URLs

1.6 Working with Tables
- 1.6.1 Basic table structure (table, thead, tbody, tr, td, th)
- 1.6.2 Attributes (colspan, rowspan)

1.7 Using HTML Forms
- 1.7.1 Creating a simple form
- 1.7.2 Understanding form elements: input, label, button
- 1.7.3 Different input types (text, email, password, etc.)
- 1.7.4 Checkboxes, radio buttons, and dropdown lists (select)

CHAPTER 2 (18 periods of practical exercises)

Multimedia, Semantic HTML and Accessibility

2.1 Adding Multimedia Elements
- 2.1.1 Embedding audio and video
- 2.1.2 Using the figure and figcaption elements

2.2 Embedding Content and iframes
- 2.2.1 Introduction to iframes
- 2.2.2 Embedding external content (maps, videos, etc.)

2.3 Adding meaning to elements with Semantic HTML
- 2.3.1 Introduction to semantic tags (article, section, header, footer)
- 2.3.2 Importance of semantic HTML in web design

2.4 HTML Quotations and Citations
- 2.4.1 Understanding the importance of citing sources
- 2.4.2 Blockquote, q, and cite tags

2.5 Understanding Accessibility in HTML
- 2.5.1 The importance of creating accessible content
- 2.5.2 Basics of web accessibility
- 2.5.3 ARIA roles and attributes

2.6 Project Work - Creating a Personal web page or web site with HTML (6 Periods)
- 2.6.1 Students should develop their own personal project, based in teacher's instructions. At the end of the project, the class and teacher will review and comment the work.

NETWORKS AND SECURITY PROTOCOLS I
(30 PERIODS)

CHAPTER 1 (3 period)

Introduction to Networking

1.1 Summary of computer networks.
1.2 Identify Terms: Servers, Clients, Hosts, etc...
1.3 Differentiate network types: LAN, WAN, MAN.
1.4 Recognize different connectors model (RJ45, Fiber Optic).

CHAPTER 2 (3 periods)

OSI Model

2.1 Brief explanation of Open System Interconnection (OSI):
2.2 Recognize 7 layers of OSI model.

CHAPTER 3 (2 periods)

Network Protocols and Parameters: (2 Periods)

3.1 Recognize protocols: IP, TCP, UDP.
3.2 Brief Explanation of Subnet Mask and Gateway.
3.3 Summary the layer protocols: HTTP, HTTPS, FTP.

CHAPTER 4 (4 periods)

Network devices and Topologies: (4 Periods)

4.1 Devices summary: routers, switches, hubs.
4.2 Summary about network topologies: Tree, Bus, Ring, Star, etc...

CHAPTER 5 (4 periods)

IP addresses and Classes: (4 Periods)

5.1 Explain IPv4 and IPv6.
5.2 Differentiate Private and Public IP.
5.3 Classes of IP addresses (Class A, Class B, Class C).

CHAPTER 6 (2 periods)

DNS and DHCP

6.1 Understanding Domain Name System (DNS).
6.2 Dynamic Host Configuration Protocol (DHCP).

CHAPTER 7 (4 periods)

Intro to Security

7.1 Summary of Network Security Concepts.

7.2 Common Network Weaknesses:

- 7.2.1 Weak Password.
- 7.2.2 Outdated Software.
- 7.2.3 Open Ports.
- 7.2.4 Open Network.
- 7.2.5 Malware and Attacks.

7.3 Importance of Encryption:

- 7.3.1 Secure Password.
- 7.3.2 Updating Software.
- 7.3.3 Authentication.
- 7.3.4 Protect from Attacks.

7.4 Basics Of Firewall.

CHAPTER 8 (3 periods)

VPN and Secure Protocols

8.1 Explain Virtual Private Network (VPN).

8.2 Recognize Secure Network Protocols (SSL / TLS).

- 8.2.1 Secure Sockets Layer. / Transport Layer Security.

8.3 Wireless computer Network Security Protocols (WPA, WPA2, etc.…)

CHAPTER 9 (2 periods)

Passwords and Authentications

9.1 Importance of strong password.

9.2 Multi-factor authentication.

CHAPTER 10 (3 periods)

Introduction to Cybersecurity and Network Management

10.1 Cybersecurity concepts, threats, and attacks.

10.2 Ethical Hacking tools and applications.

10.3 CISCO and Network devices concepts.

10.4 Discussions and Questions Related to the course.

SECOND YEAR

SPECIALIZATION SUBJECTS

1. OPERATING SYSTEMS II
2. OBJECT ORIENTED PROGRAMMING II
3. NETWORKS II
4. TP-NETWORKS
5. NETWORKS AND SECURITY
6. CLIENT SERVER ARCHITECTURE
7. VIRTUALIZATION II
8. CLOUD COMPUTING II
9. SCRIPTING AND AUTOMATION II
10. REMOTE MAINTAINANCE II
11. END OF STUDIES' PROJECT AND INTERNSHIP

OPERATING SYSTEM II
(180 PERIODS OF PRACTICAL EXERCISES)

Objective of the course

At the end of this course, the student should be able to:

- Install and configure a Windows and Linux server, manage user accounts (create users, create user groups, ensure user groups, guarantee and revoke access privileges on various files and resources for various users and user groups).
- Manage domains and domain users.
- Manage the Security and Workgroup Management

CHAPTER 1 (90 Periods of pratical exercises)

Windows Server Structure

1.4 Basic concepts of Windows (Windows rewiev TS1) and introduction to Windows Server
- 1.4.1 Windows architecture
- 1.4.2 Introduction to Windows Server and its history.
- 1.4.3 Basic concepts of the Windows Server operating system.
- 1.4.4 Introduction to management through graphical interface and command line.
- 1.4.5 Installation and initial configuration of Windows Server.
- 1.4.6 PowerShell Commands for Server Management:
  - 1.4.6.1 `GetHelp`: PowerShell help.
  - 1.4.6.2 `GetService`, `StartService`, `StopService`: Service management.
  - 1.4.6.3 `GetProcess`, `StopProcess`: Process management.
  - 1.4.6.4 `GetWindowsFeature`, `InstallWindowsFeature`, `UninstallWindowsFeature`: Windows Server feature management.

1.5 Active Directory Domain Services
- 1.5.1 Implementation of an Active Directory domain controller.
- 1.5.2 Configuration of domain services such as DNS and DHCP.
- 1.5.3 Management of Active Directory objects.
- 1.5.4 Backup and restore of Active Directory.
- 1.5.5 PowerShell Commands for Active Directory Management:
  - 1.5.5.1 `InstallWindowsFeature Name ADDomainServices`: Active Directory role installation.
  - 1.5.5.2 `AddWindowsFeature Name DNS`: DNS role installation.
  - 1.5.5.3 `AddWindowsFeature Name DHCP`: DHCP role installation.
  - 1.5.5.4 `NewADForest`, `NewADDomain`, `SetADForest`, `SetADDomain`: Active Directory domain configuration.
  - 1.5.5.5 `BackupADDSForest`, `RestoreADDSForest`: Active Directory backup and restore.

1.6 User and Group Management
- 1.6.1 Creation and management of user accounts (Local and Domain).
- 1.6.2 Configuration of account security policies.
- 1.6.3 Implementation of groups and roles.
- 1.6.4 Access rights management.
- 1.6.5 PowerShell Commands for User and Group Management:
  - 1.6.5.1 `NewADUser`, `GetADUser`, `SetADUser`, `RemoveADUser`: User management.

1.6.5.2 `NewADGroup`, `GetADGroup`, `SetADGroup`, `RemoveADGroup`: Group management.
1.6.5.3 `AddADGroupMember`, `RemoveADGroupMember`: Group member management.

1.7 File System and Permissions Management
- 1.7.1 Creation and configuration of network shares.
- 1.7.2 Assignment of permissions on files and folders.
- 1.7.3 Implementation of disk quotas.
- 1.7.4 Monitoring of storage resources.
- 1.7.5 PowerShell Commands for File System and Permissions Management:
  - 1.7.5.1 `NewSMBShare`, `GetSMBShare`, `SetSMBShare`, `RemoveSMBShare`: SMB share management.
  - 1.7.5.2 `NewItem`, `CopyItem`, `MoveItem`, `RemoveItem`: File and folder management.
  - 1.7.5.3 `GetAcl`, `SetAcl`: Permissions management.

1.8 Security and Workgroup Management
- 1.8.1 Implementation of group policies.
- 1.8.2 Configuration of Windows firewalls.
- 1.8.3 Monitoring and auditing of security activities.
- 1.8.4 Management of workgroups and nondomain servers.
- 1.8.5 PowerShell Commands for Security and Workgroup Management:
  - 1.8.5.1 `NewGPO`, `GetGPO`, `SetGPO`, `RemoveGPO`: Group Policy management.
  - 1.8.5.2 `EnableNetFirewallRule`, `DisableNetFirewallRule`: Firewall rule management.
  - 1.8.5.3 `GetWinEvent`, `GetEventLog`: Security event monitoring.
  - 1.8.5.4 `SetNetConnectionProfile`: Network profile configuration.

1.9 Troubleshooting network communications
- 1.9.1 Overview
- 1.9.2 Troubleshooting network communication with (Ping and PowerShell)
- 1.9.3 Testing connectivity to specific ports using (Command prompt and PowerShell)

1.10 Servers and DMZ
- 1.10.1 Importance and benefits of multi servers in computer network.
- 1.10.2 Servers Types (Web, Mail, DHCP, DNS, File, Print, and Database etc.)
- 1.10.3 Servers Roles and Features
- 1.10.4 Differences and Similarities between some servers
- 1.10.5 Definition and purpose of DMZ network
- 1.10.6 Importance and benefits of DMZ
- 1.10.7 DMZ Architecture and Design (Single and Dual firewall)
- 1.10.8 DMZ environment examples.

1.11 Install and Configure Web Server IIS
- 1.11.1 Importance and benefits of Web Server.
- 1.11.2 Installing, adding server roles and features
- 1.11.3 Configure IIS using server manager
- 1.11.4 Creating and uploading Website files, Virtual directories
- 1.11.5 Configuring Certificates, logging and monitoring

1.12 Managing print services

1.12.1 Install and Configure network printer driver using (Management Console and PowerShell)
1.12.2 Manage printer permissions and Queues
1.12.3 Publish a printer in Active directory using (Management Console and PowerShell)

PRACTICAL PROJECT

Implementation of a practical project involving the installation, configuration, and management of a Windows server. (Students can work on individual or group projects, with the support of the teacher)

Tasks:

1. Server installation
2. Deploy and configure various servers (Web, File, Mail, etc.)
3. User and Group Management
4. Monitoring
5. Logging in
6. Troubleshooting
7. Backing up
8. Documentation
9. Presentation

CHAPTER 2 (90 Periods of pratical exercises)

Linux Server Structure

2.1 Basic concepts of the command line (linux rewiev TS1) and introduction to Linux server structure.

- 2.1.1 File and directory management.
  - 2.1.1.1 `ls`, `cd`, `pwd`, `man`: Basic shell commands.
  - 2.1.1.2 `cat`, `less`, `head`, `tail`: Viewing file contents.
  - 2.1.1.3 `mkdir`, `rmdir`, `touch`: Managing directories and files.
  - 2.1.1.4 `cp`, `mv`, `rm`: Copying, moving, removing files and directories.
  - 2.1.1.5 `chmod`, `chown`: Modifying file permissions and ownership.
- 2.1.2 Text editor usage (vi, nano, etc.).
- 2.1.3 Fundamentals of shell scripting.

2.2 Installation and System Configuration

- 2.2.1 Installation of a Linux distribution.
- 2.2.2 Initial system configuration.
- 2.2.3 Package management and system updates.
  - 2.2.3.1 `apt`, `aptget`, `dpkg`: Package management on Debian/Ubuntu systems.
  - 2.2.3.2 `yum`, `dnf`: Package management on Red Hat/Fedora/CentOS systems.
- 2.2.4 User and group management.
- 2.2.5 Basic system security: firewall, SELinux/AppArmor, SSH.
  - 2.2.5.1 `hostnamectl`, `ifconfig`, `ip`: Network management and system configuration.
  - 2.2.5.2 `systemctl`: Control of system services (systemd).

2.3 System Administration

- 2.3.1 Process and service management.
  - 2.3.1.1 `ps`, `top`, `htop`: Process viewing and resource monitoring.
  - 2.3.1.2 `kill`, `killall`, `pkill`: Process termination.
- 2.3.2 System resource monitoring.
- 2.3.3 Task scheduling (cron, at).
- 2.3.4 Disk and partition management.
  - 2.3.4.1 `cron`, `at`: Task scheduling.

2.3.4.2 `df`, `du`: Disk space and partition management.

2.4 Networking and Network Services

2.4.1 Network configuration: IP addressing, routing, DNS, DHCP.

2.4.1.1 `ping`, `traceroute`, `netstat`: Network connectivity testing and connection viewing.

2.4.2 Configuration of the Apache server for website management.

2.4.3 Configuration of the FTP (File Transfer Protocol) server.

2.4.4 Configuration of the SSH server for secure remote access.

2.4.4.1 `iptables`, `ufw`: Firewall configuration.

2.4.4.2 `sshd`: SSH server configuration.

2.4.4.3 `httpd`, `nginx`: Web server configuration.

2.5 Database Service Management

2.5.1 Introduction to databases: MySQL/MariaDB, PostgreSQL.

2.5.2 Installation and configuration of a MySQL/MariaDB server.

2.5.2.1 `mysql`, `mysqladmin`: MySQL database management.

2.5.2.2 Creation and management of databases and users.

2.5.2.3 `psql`: Interaction with the PostgreSQL database.

2.5.2.4 Database backup and restore.

2.5.2.5 `mysqldump`, `pg_dump`: Database backup.

2.6 Virtualization and Containers

2.6.1 Basics of virtualization.

2.6.2 Installation and configuration of VirtualBox.

2.6.2.1 `VirtualBox`: Virtual machine management.

2.6.3 Creation and management of virtual machines.

2.6.4 Introduction to Docker and containerization.

2.6.4.1 `docker`: Docker container management.

2.7 Advanced Automation and Scripting

2.7.1 Automation of system administration tasks.

2.7.2 Advanced shell scripting.

2.7.2.1 Bash scripting: Advanced usage of loops, conditions, functions, variables.

2.7.3 Introduction to Ansible for configuration management.

2.7.3.1 `grep`, `sed`, `awk`: Filtering and manipulation of text data.

2.8 Advanced Security and Troubleshooting

2.8.1 Advanced Linux system security.

2.8.2 Analysis and resolution of common system problems.

2.8.2.1 `sudo`, `su`: Management of privileged access.

2.8.3 Introduction to best practices for server security.

2.8.4 Advanced management of file and directory permissions.

2.9 Practical Project

2.9.1 Implementation of a practical project involving the installation, configuration, and management of a Linux server. (Students can work on individual or group projects, with the support of the teacher).
Students can complete this project with the collaboration of the teacher of network and network security

2.10 Linux System Administration Concepts

2.10.1 Administrative Tasks

2.10.1.1 Local Server Settings Configuration (`/etc/`, `systemctl`)

2.10.1.2 Adding roles and features (`apt`, `yum`, `dnf`, or `pacman`)
2.10.1.3 Managing other servers (ssh or `rsync`, `scp`, and `sshfs`)
2.10.1.4 Creating server groups (Ansible, Puppet)
2.10.2 Overview of Server Roles and Features
2.10.2.1 Adding and Removing Server Roles and Features using Command Line
2.11 Troubleshooting network communications
2.11.1 Overview
2.11.2 Troubleshooting network communication with (`ping`, `ping -c`, and `ping -s`)
2.11.3 Testing connectivity to specific ports using Command line (`telnet`, `nc` (netcat), or `ncat`)
2.12 Servers and DMZ
2.12.1 Importance and benefits of multi servers in computer network.
2.12.2 Servers Types (Web, Mail, DHCP ISC, DNS BIND, File Samba, Print CUPS, and Database etc.)
2.12.3 Servers Roles and Features
2.12.4 Differences and Similarities between some servers
2.12.5 Definition and purpose of DMZ network
2.12.6 Importance and benefits of DMZ
2.12.7 DMZ Architecture and Design (Single and Dual firewall)
2.12.8 DMZ environment examples.
2.13 Install and Configure Apache Web Server
2.13.1 Importance and benefits of Apache Web Server.
2.13.2 Updating packages, Installing, enabling and disabling services (`sudo`)
2.13.3 Configure firewall and secure Apache
2.13.4 Creating and uploading Website files (`/var/www/html/`)
2.13.5 Logging, Testing and monitoring (`curl`, `tail`, etc.)
2.14 Managing print services
2.14.1 Install and Configure network printer driver using CUPS or Web interface (`lpadmin`, and `lpoptions`)
2.14.2 Manage printer permissions and Queues
2.14.3 18.3 Publish and manage printer sharing using Samba ((`smb.conf`, `smbpasswd`)

PRACTICAL PROJECT

Implementation of a practical project involving the installation, configuration, and management of a Linux server. (Students can work on individual or group projects, with the support of the teacher)

Tasks:

1. Server installation
2. Deploy and configure various servers (Web, File, Mail, etc.)
3. User and Group Management
4. Monitoring
5. Logging in
6. Troubleshooting
7. Backing up
8. Documentation
9. Presentation

OBJECT ORIENTED PROGRAMMING II
(120 PERIODS OF WHICH 90 ARE PRACTICAL EXERCISES)

CHAPTER 1: (25 Periods of which 15 are practical exercises)
Advanced Object-Oriented Programming in Java

1.5 Review of OOP basics
1.6 Advanced OOP features: interfaces, abstract classes, nested classes.
1.7 Design patterns and their application in Java: Factory Method, Singleton, Observer, Strategy, and Decorator.
1.8 Reflection and annotations.
1.9 Suggested practice:
- 1.9.1 Implementing various design patterns in Java.
- 1.9.2 Projects that require the use of advanced OOP concepts and design patterns.

CHAPTER 2: (25 Periods of which 15 are practical exercises)
Java File I/O and NIO

2.1 Java I/O
2.2 Review of Java I/O: FileInputStream, FileOutputStream, FileReader, and FileWriter, the File class
- 2.2.1 Review of Streams, Buffered Streams and serialization

2.3 Java New I/O (NIO) Introduction to the NIO package: Channels, Buffers, Selectors, and Path.
- 2.3.1 The advantages of NIO over traditional I/O, including non-blocking I/O operations and improved performance with large files.
- 2.3.2 Working with Files in NIO
- 2.3.3 Path, Files, and Channels API for file operations.
- 2.3.4 Exploring file attributes, directory operations, and file system monitoring.
- 2.3.5 Understanding non-blocking I/O with selectors and sockets for scalable network applications.
- 2.3.6 Introduction to asynchronous I/O with CompletableFuture and AsynchronousFileChannel.
- 2.3.7 Suggested practice: developing a file search utility using NIO features.

CHAPTER 3: (20 Periods of which 15 are practical exercises)
Concurrency in Java

3.1 Introduction to Concurrency:
- 3.1.1 Understanding the need for concurrency and its impact on software development.
- 3.1.2 Basics of processes and threads: the difference between them, how they work in Java.

3.2 Threads in Java:
- 3.2.1 Creating threads using the Thread class and the Runnable interface.
- 3.2.2 Thread lifecycle, states, and transitions.

3.3 Synchronization and Thread Safety:
- 3.3.1 The concept of thread safety and methods to achieve it in Java.

3.4 Synchronized methods and blocks, the volatile keyword.
3.5 Overview of the java.util.concurrent package.

3.5.1 Executors, ExecutorService, and Thread Pools for managing a collection of threads.
3.5.2 Callables and Futures for result-bearing tasks.
3.6 Locks and Synchronization Utilities:
3.6.1 Lock objects for controlling access to a resource by multiple threads.
3.6.2 Condition objects for inter-thread communication.
3.7 Introduction to thread-safe collections in Java.
3.7.1 ConcurrentHashMap, CopyOnWriteArrayList, BlockingQueue, and their use cases.
3.8 Parallel Streams and Fork/Join Framework:
3.8.1 Leveraging parallel streams for simplifying parallelism in Java applications.
3.8.2 Understanding the Fork/Join framework for recursive task parallelism.
3.9 Asynchronous Programming in Java:
3.9.1 CompletableFuture for asynchronous programming and its advantages over Future.
3.9.2 Using the CompletableFuture API for composing asynchronous logic.
3.10 Suggested practice:
3.10.1 Exercises on creating, running, and controlling threads.
3.10.2 Utilizing the Fork/Join framework for tasks that can be broken down into smaller parts.

CHAPTER 4: (30 Periods of which 25 are practical exercises)

Java Networking and Web Programming

4.1 Review of networking basics
4.2 Overview of web application components: Servlets, JSP, and the MVC architecture.
4.3 Exploring Java EE technologies: JSF, JPA for persistence, EJB for business logic.
4.4 Introduction to Spring Framework: Core, MVC, Security, and Boot.
4.5 Planning, developing, testing, and deploying web applications in Java.
4.6 Understanding web servers (Tomcat, Jetty) and application servers (WildFly, GlassFish).
4.7 Introduction to microservices architecture with Spring Boot.
4.8 Basic concepts of web security.
4.9 Implementing authentication and authorization in Java web applications.

CHAPTER 5: (20 Periods of practical exercises)

Final project

Students will work individually or in small teams to develop a final project that demonstrates their understanding of Java concepts.
Possible simple applications students could develop include:

- **E-commerce Platform**: A web-based e-commerce application where users can browse products and add them to a cart.
  User authentication and session management should be implemented.
- **Chat Application**: A real-time chat application using Java networking (sockets) that supports multiple clients connected to a server. It could include features like private and group chats, file transfers, and user statuses (online, offline).

NETWORKS II

(90 PERIODS)

Description of the material

This course is a continuation of the network course given in the 1st year of TS Systems and Networks. It is divided into two parts. The first part deals with interconnection and redundancy in networks. The second deals in depth and in detail with the TCP/IP protocol

Objective of the subject

At the end of the course, the student will be able to:

- Identify the elements of a local or wide area network.
- Design and schematically represent local networks.
- Design and schematically represent VLANs
- Configure some equipment.
- Use network monitoring and control tools.
- Check the TCP/IP protocol.

Skills and Abilities (Learning Outcomes)

At the end of this course, the student should be able to analyze, design, identify the elements of a network and configure a local area network and be familiar with the advanced notions of the network concept.

CHAPTER 1 (6 Periods)

Network Recap

1.1 Open System Interconnection (ISO) Reference Model
1.2 Packages: definition, creation (header)
1.3 The protocols of the different layers of ISO.
1.4 The architecture of the Internet
1.5 Point to point
1.6 ethernet
1.7 IP addressing (classes A, B,…)
1.8 Reserved IPs
1.9 Masking.
1.10 Others….
1.11 Application exercises

CHAPTER 2 (8 Periods)

Network Interconnections

2.1 Introduction.
2.2 LAN-LAN
  2.2.1 Definition
  2.2.2 Examples and application exercises (number of server stations, number of client stations, topology, cabling (type and length), protocol, addresses, interconnection equipment, etc.)
2.3 LAN-WAN.
  2.3.1 Definition
  2.3.2 Routing
    2.3.2.1 Direct and Indirect routing
    2.3.2.2 The routing process and table
    2.3.2.3 Routing protocols (interior v/s exterior)
      2.3.2.3.1 Troubleshooting Routing Information Protocol (RIP)

2.3.2.3.2 Troubleshooting Open Shortest Path First (OSPF)
2.3.2.3.3 Troubleshooting Enhanced Interior Gateway Routing Protocol (EIGRP)
2.3.3 Examples
2.4 WAN - WAN.
2.4.1 Definition
2.4.2 Exterior Gateway Protocol
2.4.2.1 BGP
2.4.3 Examples
2.5 Interconnection functions
2.5.1 Issue of different protocols: list the cases.
2.5.2 Error handling.
2.6 Application
2.6.1 The commands (ipconfig, ipconfig /all, traceroute, ping, road, netconfig, and netstat)

CHAPTER 3 (8 Periods)

Redundancy in networks

3.1 Definition and necessity of redundancy.
3.2 Redundancy architecture at equipment and link level.
3.2.1 Standard RAID Levels
3.2.1.1 RAID 0, RAID 1, RAID 5, RAID 6, RAID 10, and RAID 0+1
3.2.1.2 Common Technologies
3.3 New storage solutions
3.3.1 Introduction
3.3.2 The Network Attached Storage NAS solution
3.3.2.1 Features
3.3.2.2 Example of a NAS server
3.3.3 The Storage Area Network SAN Solution
3.3.3.1 Features
3.3.3.2 Benefits
3.3.4 SAN vs. NAS
3.4 Recovery
3.4.1 Fault detection.
3.4.2 Switch to the fallback solution.
3.4.3 Back to normal environment.

CHAPTER 4 (30 Periods)

TCP/IP networks

4.1 How TCP/IP Networks Work
4.1.1 Synchronization issues
4.1.2 IPv4 and IPv6 addressing
4.1.3 ARP/RARP: Address resolution protocols
4.1.4 The IP architecture (the header)
4.1.5 The TCP architecture (the header)
4.1.6 Applications that use TCP/IP (Telnet, FTP, SMTP)
4.1.7 IP addressing exercises (VLSM and CIDR)

4.2 IP: Internet Protocol
- 4.2.1 IPv4
- 4.2.2 IPv6
- 4.2.3 Conversion from IPv4 to IPv6

4.3 UDP: User Datagram Protocol unreliable transport service
- 4.3.1 Definition
- 4.3.2 On your mind
- 4.3.3 Applications that use UDP (NFS, SNMP, DNS)

4.4 TCP and UDP ports
- 4.4.1 TCP: a reliable transport service (+ Exercises)
- 4.4.2 IPs mobile
- 4.4.3 The allocation of TCP/IP addresses and parameters
  - 4.4.3.1 BOOT strap protocol
  - 4.4.3.2 DHCP (Dynamic Host Configuration Protocol)
  - 4.4.3.3 Functioning
  - 4.4.3.4 implementation
  - 4.4.3.5 on your mind

4.5 ICMP: Internet Control Message Protocol
- 4.5.1 error messages in IPv4

4.6 IP telephony

4.7 DNS
- 4.7.1 definition
- 4.7.2 hierarchy
- 4.7.3 components
  - 4.7.3.1 nameserver
  - 4.7.3.2 nameresolver
- 4.7.4 DNS server types
  - 4.7.4.1 rootserver
  - 4.7.4.2 primaryormasterserver
  - 4.7.4.3 secondaryorslaveserver
- 4.7.5 DNS records
  - 4.7.5.1 Internet(IN)
  - 4.7.5.2 NameServer(NS)
  - 4.7.5.3 StartofAuthority(SOA)
  - 4.7.5.4 Address(A, YYYY)
  - 4.7.5.5 CanonicalName(CNAME)
  - 4.7.5.6 MailExchanger(MX)
  - 4.7.5.7 Point(PTR)
  - 4.7.5.8 TextTXT
- 4.7.6 Application exercises

CHAPTER 5 (20 Periods)

Local Virtual Networks - VLAN

5.1 Introduction

5.2 Interest of VLANs

5.3 Reminder on the operation of the switches

5.4 Algorithm of spanning tree (SpanningTree – STP)

5.5 VLAN Example

CHAPTER 6 (16 Periods)

Internet/Intranet networks

- 6.1 Internet 1
  - 6.1.1 Historical
  - 6.1.2 Standardization status
- 6.2 Internet 2
- 6.3 Areas of use
- 6.4 Topology
- 6.5 ISP
- 6.6 Internet applications
  - 6.6.1 Classic applications
    - 6.6.1.1 SMTP (Simple Mail Transfer Protocol)
    - 6.6.1.2 FTP (File Transfer Protocol)
    - 6.6.1.3 net
    - 6.6.1.4 NFS (Network File System)
    - 6.6.1.5 WWW
  - 6.6.2 Multipoint applications
- 6.7 Intranet
  - 6.7.1 Definition
  - 6.7.2 Intranet network utilities.
  - 6.7.3 Areas of use.
  - 6.7.4 Network management.
  - 6.7.5 Topology
  - 6.7.6 Examples.
- 6.8 Extranet
  - 6.8.1 Definition.
  - 6.8.2 Extranet network utilities.
  - 6.8.3 Areas of use.
  - 6.8.4 Network management.
  - 6.8.5 Topology.
  - 6.8.6 Examples.
- 6.9 Infranet.
  - 6.9.1 Definition.
  - 6.9.2 Infranet network utilities.
  - 6.9.3 Areas of use.
  - 6.9.4 Network management.
  - 6.9.5 Topology.
  - 6.9.6 Examples.
- 6.10 Future market trends : Internet 3, …

CHAPTER 7 (4 Periods)

Access networks

- 7.1 Hertzian access
  - 7.1.1 Wireless local loop
  - 7.1.2 All generations of mobile networks (First, Second, …)
  - 7.1.3 Generations of wireless communications networks

- 7.1.4 Broadband wireless access
- 7.1.5 Satellite access

7.2 Mobile-IP
- 7.2.1 Nodes (Mobile, Home, Foreign, Tunneling, and Security)
- 7.2.2 Addressing (HomeAddress, CareOfAddress)

7.3 Wireless LANs: IEEE 802.11

7.4 The architectures
- 7.4.1 Access techniques
- 7.4.2 Interfaces

7.5 Extensions
- 7.5.1 GPRS
- 7.5.2 UMTS
- 7.5.3 WAP
- 7.5.4 IEEE 802.15
- 7.5.5 ad hoc networks
  - 7.5.5.1 Definition

TP-NETWORK

(90 PERIODS OF PRACTICAL EXERCISES)

Description of the material

The Computer Networks Lab is specially set up for labs and network projects and includes Ethernet or Token Ring networks. The environment is not necessarily homogeneous both in terms of platforms (Windows, Netware, Linux) or network equipment (hubs, switches, bridges, routers). Studies are performed using test tools, simulation software and protocol analyzers.

Objective of the subject

At the end of the project, the student will be able to demonstrate their ability to analyze and develop:

- Local networks with Ethernet equipment, TokenRing, etc.
- TCP/IP protocols and protocol developments;
- Connection to other laboratories via multi-protocol routers;
- Studies using test tools, simulation software and protocol analyzers;
- Administration techniques for heterogeneous networks (IP, IPv6, etc.);
- Network audits: architectures, protocols;
- Network management;
- Securing networks;
- Suggest to students to seek innovative solutions following problems created by the professor allowing them to anticipate strategic developments,

Work to do :

A- Step 1 – Preliminary, carried out by the laboratory manager (6 Periods):

1. Presentation of the different elements of a LAN network (server, client, router, concentrator, etc.)
2. Demonstration of a network analyzer on an existing functional network (congestion, statistics, connection, packet circulation, failure detection, reconfiguration, return to normal, etc.)

B- Step 2 – Analysis of simple cases, carried out by group (6 Periods):

1. Identify the needs of each given case study and represent them on a diagram (equipment, cabling, topology, operating system, etc.).
2. Installation of units (server, stations, network card or modem, cabling, concentrator, etc.).
3. Verification of network operation

C- Step 3 – Practical work covering the following themes (carried out by group) (60 Periods)

1. Orientation of the topology and creation of a small network
2. Using Wireshark to display protocol data units, parsing IP packets

3. Examine an ICMP packet
4. Supervision of the operation of a network (traceroute, ping, IP scanning, port scanning, etc.)
5. Configuring Basic Router Settings
6. Configuring and Troubleshooting a Static Route
7. Configuring and troubleshooting RIPv1 and RIPv2 (VLSM and CIDR)
8. EIGRP configuration and troubleshooting
9. OSPF configuration and troubleshooting
10. VLAN configuration and troubleshooting
11. NAT and DHCP configuration and troubleshooting

D- Step 4 – Security, carried out by group (18 hours):

D.1: Access List

1. Configuration and troubleshooting Access List (standard, extended, named)
2. Hardware-level implementation of a router.
3. Implementing security on a network. (Allow and/or deny access via IP on server, hub and router)

Materials:

Please find below the minimum material required for each group which is composed of a maximum of three students.

For each group

1. Server (containing 2 network cards)
2. 2 client stations (1 network card)
3. Switch (SWITCH)
4. Cabling (50 and 75 ohm coaxial cable, twisted pair (STP, UTP, 4 and 8 lines), Optical fiber (Laser, Infrared, LED)
5. Connectors (RJ45, Terminators,….)
6. Network cable tester.
7. Network oscilloscope (to validate the signal traffic on the cable)
8. Modem
9. Print Server
10. Connection clamp for RJ45 and RJ11

For the lab:

1. Network analyzer (Network Analyzer ex: Fluke)
2. Network monitoring software (What's up gold, SMTP, etc.)
3. Switch

4. Router
5. Barrier protection (FIREWALL)
6. 2 main servers
7. Smart-UPS (RJ45)
8. Printer
9. Modem
10. Internet connection

Simulator: (Choose one or more)

- Packet Tracer
- Boson NetSim
- GNS
- Others …

NETWORK SECURITY
(120 PERIODS OF WHICH 90 ARE PRACTICAL EXERCISES)

CHAPTER 1 (20 Periods of which 10 are practical exercises)

Network Security Recap

1.1 Review of Concepts:
- 1.1.1 Networking terms, types, connectors, and models.
- 1.1.2 Protocols, parameters, devices, and topologies.
- 1.1.3 IP addresses, classes, DNS, and DHCP.

1.2 Transmission:
- 1.2.1 Elements, Media and Techniques.

1.3 Cybersecurity:
- 1.3.1 Importance.
- 1.3.2 Threats: (Malware, Phishing, DoS, MITM…).
- 1.3.3 Security Vulnerability: (Hardware, Software, and Updates).
- 1.3.4 Privacy Protection: (Encryption, Multi factor…).
- 1.3.5 Roles and Technologies: (IDS, IPS, VPN…).

1.4 Security Types, Benefits, and Challenges:
- 1.4.1 Types: (Antivirus, Antimalware, Cloud Security, Data Loss, Email, Firewall…).
- 1.4.2 Benefits: (Functionality, Confidentiality, Integrity and Availability. Intellectual property protection).
- 1.4.3 Challenges: (Remote access, User adherence…).

CHAPTER 2 (35 Periods of which 30 are practical exercises)

Attacks, Security Concepts and Techniques

2.1 Analyzing a Cyber Attack
- 2.1.1 Types of Malware
- 2.1.2 Symptoms of Malware

2.2 Methods of Infiltration
- 2.2.1 Social Engineering
- 2.2.2 Denial-of-Service
- 2.2.3 Distributed DoS
- 2.2.4 Botnet
- 2.2.5 On-Path Attacks
- 2.2.6 SEO Poisoning
- 2.2.7 Wi-Fi Password Cracking
- 2.2.8 Password Attacks
- 2.2.9 Cracking Times
- 2.2.10 Advanced Persistent Threats

2.3 Security Vulnerability and Exploits
- 2.3.1 Hardware Vulnerabilities
- 2.3.2 Software Vulnerabilities
- 2.3.3 Categorizing Software Vulnerabilities
- 2.3.4 Software Updates

2.4 The Cybersecurity Landscape
- 2.4.1 Cryptocurrency
- 2.4.2 Cryptojacking

CHAPTER 3 (35 Periods of which 30 are practical exercises)

Protecting Data and Privacy

3.1 Protecting Your Devices and Network
- 3.1.1 Protecting Your Computing Devices
- 3.1.2 Wireless Network Security at Home
- 3.1.3 Public Wi-Fi Risks
- 3.1.4 Password Security
- 3.1.5 A Strong Password
- 3.1.6 Using a Passphrase
- 3.1.7 Password Guidelines
- 3.1.8 Password Check

3.2 Data Maintenance
- 3.2.1 What Is Encryption?
- 3.2.2 How Do You Encrypt Your Data?
- 3.2.3 Back Up Your Data
- 3.2.4 Are They Really Gone?
- 3.2.5 How Do You Delete Your Data Permanently?

3.3 Who Owns Your Data?
- 3.3.1 Terms of Service
- 3.3.2 Understand the Terms
- 3.3.3 What Are You Agreeing To?
- 3.3.4 The Data Use Policy
- 3.3.5 Privacy Settings
- 3.3.6 Before You Sign Up
- 3.3.7 Protect Your Data

3.4 Safeguarding Your Online Privacy
- 3.4.1 Two Factor Authentication
- 3.4.2 Open Authorization
- 3.4.3 Social Sharing
- 3.4.4 Don't Get Spoofed
- 3.4.5 Email and Web Browser Privacy

CHAPTER 4 (30 Periods of which 20 are practical exercises)

Protecting the Organization

4.1 Cybersecurity Devices and Technologies
- 4.1.1 Security Appliances
- 4.1.2 Firewalls
- 4.1.3 Port Scanning
- 4.1.4 What Does It Mean?
- 4.1.5 Intrusion Detection and Prevention Systems
- 4.1.6 Real-Time Detection
- 4.1.7 Protecting Against Malware
- 4.1.8 Security Best Practices

4.2 Behavior Approach to Cybersecurity
- 4.2.1 Behavior-Based Security
- 4.2.2 NetFlow
- 4.2.3 Penetration Testing
- 4.2.4 Impact Reduction

4.2.5 What Is Risk Management?

CLIENT SERVER ARCHITECTURE
(60 PERIODS)

Objective of the Topic

- Design and develop the C/S Components and models.
- Administration of C/S applications.
- Understand mediators and remote procedural calls.

CHAPTER 1 (10 Periods)

Introduction to C/S Architecture:

1.1 Overview of C/S
  1.1.1 Brief History
  1.1.2 Definition
1.2 Network Basics and Protocols.
1.3 Security and Authentications.
1.4 Difference between client-side and server-side architecture.

CHAPTER 2 (20 Periods)

C/S Architecture Examples & Components:

2.1 C/S Architecture (***WEB and Network***) Examples
  2.1.1 Mail Servers, File Servers, and WEB Servers.
  2.1.2 Banking Application
  2.1.3 Healthcare, Banking, and Airport Application
2.2 Components of C/S Architecture
  2.2.1 Workstations (Client-side)
  2.2.2 Servers (Mail, File, DNS, and WEB, etc.)
  2.2.3 Networking Devices (Hubs, Switches, Repeaters, and Bridges, etc.)

CHAPTER 3 (20 Periods)

Working Steps and Types for a C/S Architecture:

3.1 How does C/S Architecture work?
3.2 C/S Architecture types and Subtypes
  3.2.1 One-tier Architecture
  3.2.2 Two-tier Architecture
  3.2.3 Three-tier Architecture
  3.2.4 N-tier Architecture
  3.2.5 Thin-Client Architecture
  3.2.6 Fat-Client Architecture
3.3 Difference Between Peer-to-Peer Network and C/S Architecture.
3.4 Advantages and Disadvantages of C/S Architecture.
3.5 Use Cases of C/S Architecture.

CHAPTER 4 (10 Periods)

Mediators:

4.1 Definition and Objectives
4.2 Basic and Architecture of Mediators.
4.3 Main features of the Mediation Service
- 4.3.1 Login procedure
- 4.3.2 Preparation of requests
- 4.3.3 Executing queries
- 4.3.4 Retrieval of results
- 4.3.5 Results cache management
- 4.3.6 Query cache management

4.4 Families of mediators
- 4.4.1 RPC middleware
- 4.4.2 Object Middleware
- 4.4.3 Message Oriented Middleware
- 4.4.4 Component Middleware
- 4.4.5 Middleware Service
- 4.4.6 Database middleware
- 4.4.7 Persistence middleware
- 4.4.8 transaction middleware

VIRTUALIZATION II

(60 PERIODS OF WHICH 30 ARE PRACTICAL EXERCISES)

CHAPTER 1 (12 Periods of which 6 are practical exercises)

Recap of Virtualization

1.1 Importance of Virtual Computing.

1.1.1 Types (Hardware, Software, etc.)

1.2 Software categories and features.

1.2.1 Overview of Hypervisors Types.

1.2.2 Installing and configuring products (VMware, Hyper-V, etc.)

CHAPTER 2 (14 Periods of which 7 are practical exercises)

Virtual environment and Networking:

2.1 Concept and Importance of Virtual network.

2.2 Functions and roles of Virtual Switches.

2.3 Explain and Explore network modes (NAT, Bridged, etc.)

CHAPTER 3 (10 Periods of which 5 are practical exercises)

Windows and Linux Virtual environment:

3.6 Difference between VM in both OS

3.6.1 Types, Management Tools, Cost, Licensing and Open source or No

3.6.2 Examples and Practice about above points

CHAPTER 4 (12 Periods of which 6 are practical exercises)

Microsoft Hyper-V Advanced concepts (Windows):

4.1 Advanced Hyper-V components and features

4.2 Hyper-V Replica between hosts.

4.3 Configure and Implement Network (Redundancy, Load Balancing, and Isolation etc.)

4.4 Configure Remote FX settings, and Advanced Security features

4.5 Practice on Configuration, Securing and Troubleshooting.

CHAPTER 5 (12 Periods of which 6 are practical exercises)

VMware Advanced concepts (Linux):

5.1 Advanced VMware components and features

5.2 Configure and Implement Network (Teaming, Load Balancing, and Segmentation)

5.3 Configure Storage technologies (MFS “Machine file system” and NFS “Network FS”)

5.4 Implement Backup and Security Encryption

5.5 Practice on Configuration, Securing and Troubleshooting.

CLOUD COMPUTING II
(120 PERIODS OF WHICH 90 ARE PRACTICAL EXERCISES)

CHAPTER 1 (30 Periods)

Understanding the Cloud

1.1 Core Cloud Concepts
- 1.1.1 Virtualization: Understand the technology that enables cloud computing, including hypervisors and virtual machines.
- 1.1.2 Storage in the Cloud: Introduction to cloud storage options, object storage, block storage, and file storage.
- 1.1.3 Networking in the Cloud: Basics of cloud networking, including virtual networks, subnets, and internet gateways.

1.2 Security and Compliance in the Cloud
- 1.2.1 Cloud Security Best Practices: The shared responsibility model, encryption, and IAM.
- 1.2.2 Compliance and Regulations: Overview of common compliance standards (e.g., GDPR, HIPAA) and how cloud providers help meet these requirements.

1.3 Cloud Monitoring and Management
- 1.3.1 Monitoring Tools: Introduction to monitoring cloud resources for performance, availability, and cost.
- 1.3.2 Alerts and Notifications: Setting up alerts for monitoring cloud resources.
- 1.3.3 Cloud Management Practices: Overview of managing cloud resources, including automation and configuration management.

1.4 Cloud Networking
- 1.4.1 Introduction to Cloud Networking: Basics of setting up and managing networks in the cloud.
- 1.4.2 Load Balancers and Traffic Management: Understand how load balancing works for distributing traffic and ensuring high availability.
- 1.4.3 Network Security: Basics of network security groups (NSGs), firewalls, and other security measures.

1.5 Database Services in the Cloud
- 1.5.1 Overview of Cloud Databases: Introduction to types of database services offered in the cloud (SQL, NoSQL).
- 1.5.2 Choosing the right Database service: Factors to consider when selecting a database service for your application.

1.6 DevOps in the Cloud
- 1.6.1 Understanding DevOps: Explore the DevOps philosophy, benefits, and its importance in cloud computing.
- 1.6.2 Continuous Integration and Continuous Deployment (CI/CD): Basics of automating the software delivery process.
- 1.6.3 Infrastructure as Code (IaC): Introduction to managing infrastructure using code for consistent and efficient deployment.

1.7 Microservices and Containers
- 1.7.1 Introduction to Microservices: Understand the microservices architecture and its benefits.
- 1.7.2 Containers and Orchestration: Learn about containerization technologies (e.g., Docker) and orchestration tools (e.g., Kubernetes).

1.8 Cloud Development Tools

1.8.1 CI/CD Tools: Overview of continuous integration and continuous deployment tools available in the cloud.
1.8.2 Infrastructure as Code Tools: Introduction to tools like Terraform and AWS CloudFormation.
1.8.3 Version Control Systems: Importance of using version control systems (e.g., Git) in cloud development.

CHAPTER 2 (30 Periods of practical exercises)

Microsoft Azure

2.1 Security and Compliance
- 2.1.1 Security Principles in MS Azure
- 2.1.2 Roles and Access Control
- 2.1.3 Compliance and Regulations
- 2.1.4 Suggested practice: configuring Azure Active Directory and Role-Based Access Control (RBAC)

2.2 Monitoring and Diagnostics
- 2.2.1 Monitoring Tools in MS Azure
- 2.2.2 Configuration of Alerts and Notifications
- 2.2.3 Suggested practice: Setting Up Monitoring and Alerts for a Web Application

2.3 Azure Networking
- 2.3.1 Virtual Networks : Configuration and Connectivity.
- 2.3.2 Load Balancers and Traffic Management
- 2.3.3 Network Security Groups (NSGs) and Application Gateway
- 2.3.4 Suggested practice:
  - 2.3.4.1 Creating and Configuring a Virtual Network and NSG
  - 2.3.4.2 Configuring Load Balancers for High Availability

2.4 Database Services in Azure
- 2.4.1 Azure SQL Database
- 2.4.2 Cosmos DB
- 2.4.3 Azure Database for MySQL and PostgreSQL
- 2.4.4 Suggested practice:
  - 2.4.4.1 Deploying and Managing an Azure SQL Database
  - 2.4.4.2 Implementing a NoSQL Database with Cosmos DB

2.5 Azure Development Tools
- 2.5.1 Azure DevOps Services: Boards, Repos, Pipelines, Artifacts.
- 2.5.2 GitHub Actions for Azure
- 2.5.3 Visual Studio Code Azure Extensions
- 2.5.4 Suggested practice:
  - 2.5.4.1 Setting Up a CI/CD Pipeline with Azure DevOps
  - 2.5.4.2 Deploying Code from GitHub to Azure using GitHub Actions

2.6 Final Project
- 2.6.1 Design, Implement, and Deploy an Application Solution using Azure Services

CHAPTER 3 (30 Periods of practical exercises)

Google Cloud Platform

3.1 Security and Compliance in GCP
- 3.1.1 Security Principles
- 3.1.2 Roles and Access Control
- 3.1.3 Compliance and Regulations
- 3.1.4 Suggested practice: managing IAM Policies and Roles

3.2 Monitoring and Management Tools
- 3.2.1 Stackdriver Monitoring: Monitoring cloud resources and applications.
- 3.2.2 Stackdriver Logging: Log management and analysis.
- 3.2.3 Stackdriver Error Reporting: Identifying and understanding errors in your applications.
- 3.2.4 Suggested practice:
  - 3.2.4.1 Configuring Monitoring and Alerts with Stackdriver
  - 3.2.4.2 Implementing Logging and Error Reporting

3.3 Networking in Google Cloud
- 3.3.1 Virtual Private Cloud (VPC): Creating isolated networks within GCP.
- 3.3.2 Cloud Load Balancing: Distributing user traffic across multiple instances.
- 3.3.3 Cloud VPN and Interconnect: Connecting on-premises networks to GCP.
- 3.3.4 Suggested practice:
- 3.3.5 Setting Up and Configuring a VPC Network
- 3.3.6 Implementing Load Balancing for High Availability

3.4 DevOps in Google Cloud
- 3.4.1 Cloud Build: Continuous integration and delivery platform.
- 3.4.2 Cloud Deployment Manager: Infrastructure as code for GCP resources.
- 3.4.3 Cloud Source Repositories: Private Git repositories hosted on GCP.
- 3.4.4 Cloud Functions: Event-driven serverless computing.
- 3.4.5 Suggested practice:
  - 3.4.5.1 Automating Deployment with Cloud Deployment Manager
  - 3.4.5.2 Building and Deploying Applications with Cloud Build
- 3.4.6 Final Project
  - 3.4.6.1 Designing, Implementing, and Deploying a Solution Using GCP Services

CHAPTER 4 (30 Periods of practical exercises)

Amazon Web Services (AWS)

4.1 AWS Security and Compliance
- 4.1.1 Advanced IAM: Policies, Groups, and Best Practices.
  Compliance on AWS: Understanding AWS Compliance Programs.
  Data Protection: Encryption, KMS, and Certificate Manager.
- 4.1.2 Suggested practice: Implementing Security Best Practices with IAM

4.2 Monitoring and Diagnostics
- 4.2.1 Amazon CloudWatch: Monitoring AWS resources and applications.
- 4.2.2 AWS CloudTrail: Governance, compliance, operational auditing, and risk auditing of your AWS account.
- 4.2.3 AWS X-Ray: Analyzing and debugging distributed applications.
- 4.2.4 Suggested practice:
  - 4.2.4.1 Setting Up Monitoring with Amazon CloudWatch

4.2.4.2 Tracing Application Requests with AWS X-Ray

4.3 Developing and Deploying Applications
- 4.3.1 CI/CD on AWS: AWS CodeCommit, CodeBuild, CodeDeploy, and CodePipeline.
- 4.3.2 AWS Developer Tools: Overview of the AWS Developer Tools Suite.
- 4.3.3 Microservices on AWS: Deploying microservices-based applications on AWS.
- 4.3.4 Container Services: Amazon ECS and EKS.
- 4.3.5 Suggested practice: Building and Deploying a Web Application Using AWS Developer Tools

4.4 Advanced Networking on AWS
- 4.4.1 Amazon Route 53: Scalable Domain Name System.
- 4.4.2 Amazon CloudFront: Content Delivery Network services.
- 4.4.3 AWS Direct Connect: Dedicated network connection to AWS.

4.5 VPC Concepts
- 4.5.1 VPC Peering
- 4.5.2 VPN Connections.
- 4.5.3 Suggested Practice: Setting Up a Content Delivery Network with Amazon CloudFront

4.6 Final Project
- 4.6.1 Designing, Implementing, and Deploying an AWS-Based Solution

SUGGESTED RESOURCES:

Azure:
Microsoft training at: https://learn.microsoft.com/en-us/training/browse/ and in particular:

- Azure networking: https://learn.microsoft.com/en-us/training/paths/design-implement-microsoft-azure-networking-solutions-az-700/
- Azure storage: https://learn.microsoft.com/en-us/training/paths/secure-compute-storage-databases/
- Azure DevOps: https://learn.microsoft.com/en-us/training/paths/evolve-your-devops-practices/

AWS:
AWS Training at https://explore.skillbuilder.aws/learn and in particular:

- AWS Basics: https://explore.skillbuilder.aws/learn/public/learning_plan/view/82/cloud-essentials-knowledge-badge-readiness-path
- AWS Advanced: https://explore.skillbuilder.aws/learn/public/learning_plan/view/1044/solutions-architect-knowledge-badge-readiness-path

Google Cloud:
Google Training at https://cloud.google.com/learn/ and in particular:

- Cloud Engineer:
- https://www.cloudskillsboost.google/paths/11
- DevOps Engineer:

- https://www.cloudskillsboost.google/paths/20

SCRIPTING AND AUTOMATION II

(90 PERIODS OF WHICH 60 ARE PRACTICAL EXERCISES)

CHAPTER 1: (10 Periods of practical exercises)

Recap of Previous Year's Program

1.1 Review of Scripting Basics

1.2 Recap of fundamental scripting concepts in Bash and PowerShell.

1.3 Best practices for script structure, debugging, and error handling.

1.4 Practical exercises revisiting basic scripting tasks in Bash and PowerShell.

CHAPTER 2: (30 Periods)

Scripting Architecture and design

2.1 Scripting Best Practices and Design Patterns

- 2.1.1 Writing Maintainable Scripts
- 2.1.2 Code organization, commenting, and documentation.
- 2.1.3 Version control best practices for scripts.
- 2.1.4 Design Patterns in Scripting
- 2.1.5 Common design patterns and architectures for automation scripts.

2.2 Security Considerations in Scripting

- 2.2.1 Securing Script Execution
- 2.2.2 Managing credentials and sensitive data in scripts.
- 2.2.3 Secure scripting techniques to prevent injection attacks and data leakage.
- 2.2.4 Scripting in Compliance and Regulatory Environments
- 2.2.5 Understanding the impact of compliance (e.g., GDPR, HIPAA) on scripting practices.
- 2.2.6 Strategies for logging, auditing, and reporting script activities in regulated environments.

2.3 Scripting and Cloud

- 2.3.1 Introduction to DevOps and Automation in the Cloud
- 2.3.2 Overview of DevOps practices and tools for automation.
- 2.3.3 Cloud-based automation tools and services (e.g., AWS Lambda, Azure Functions).
- 2.3.4 Emerging Scripting Languages and Tools

CHAPTER 3: (15 Periods of practical exercises)

Advanced Bash Scripting

3.1 Bash Scripting for Linux

- 3.1.1 Text Processing and Sed/Awk
- 3.1.2 Advanced text manipulation tasks using sed and awk.
- 3.1.3 Incorporating external programs and multi-file scripts.
- 3.1.4 Suggested practice: Creating scripts for report generation and data extraction.

3.2 Bash Scripting for Network Administration

- 3.2.1 Using Bash for network diagnostics and configuration.

3.3 Suggested practice: Writing scripts to automate network scanning and configuration tasks.

CHAPTER 4: (15 Periods of practical exercises)

Advanced PowerShell Scripting

4.1 Working with APIs and Web Services

4.2 Automating interactions with RESTful APIs using PowerShell.

4.3 PowerShell DSC (Desired State Configuration)

4.4 Managing and maintaining Windows infrastructure using DSC.

4.5 Suggested practice:

- 4.5.1 Scripting solutions to integrate with web services and cloud APIs.
- 4.5.2 Developing DSC configurations for automated system setup and compliance.

4.6 Advanced Error Handling and Debugging

- 3.1 Deep dive into robust error handling and debugging techniques.

4.7 Suggested practice: Enhancing scripts with comprehensive error handling and diagnostic logging.

CHAPTER 5: (20 Periods of practical exercises)

Introduction to Python for Automation

5.1 Introduction to Python for Automation

- 5.1.1 Python Basics for Scripting
- 5.1.2 Introduction to Python syntax and setting up the environment.

5.2 Suggested practice: Writing simple Python scripts for file and directory management.

5.3 Automating Administrative Tasks with Python

- 5.3.1 Using Python for system administration, network monitoring, and configuration.
- 5.3.2 Developing Python scripts for log analysis, system health checks, and automated reporting.
- 5.3.3 Leveraging Python Libraries for Automation

REMOTE MAINTAINANCE II
(30 PERIODS OF PRACTICAL EXERCISES)

CHAPTER 1 (10 Periods)

Recap of Remote Maintenance

1.1 Review of concepts
- 1.1.1 Importance, problems and solutions.
- 1.1.2 RDP, SSH, Authentication and Tunneling.
- 1.1.3 TeamViewer or other tool (Benefits, Practical and Troubleshooting)
- 1.1.4 Remote Policies (Security, Protection, Encryption and Authentication)

CHAPTER 2 (12 Periods)

RUST Setup and Technologies

2.1 Overview of Configuring Remote Desktop Protocol.

2.2 RUST Tools:
- 2.2.1 Introduction and Benefits.
- 2.2.2 Comparing RUST with other Remote Tools (Features, Advantages and Disadvantages).
- 2.2.3 Install and Configure RUST on different platforms (Mac, Windows, Linux).
- 2.2.4 RUST Architecture and Components analysis.
- 2.2.5 Practical Exercises for RDP, SSH and RUST.
- 2.2.6 Troubleshooting and Q&A.

2.3 Configuring Remote Access Policies.

CHAPTER 3 (8 Periods)

RUST Security and Solutions

3.1 Security Overview and Importance
- 3.1.1 User Authentication.
- 3.1.2 Device Security.

3.2 Data Protection and Encryption.
- 3.2.1 Secure Data.
- 3.2.2 Encryption Overview and Role.

3.3 Identifying Issues and Solutions.

3.4 Plan and Develop a RUST project.

END OF STUDIES' PROJECT AND INTERNSHIP
(180 PERIODS)

END OF STUDIES' PROJECT (120 Periods)

Description of the material

This course is a design project designed to allow students to put into practice the concepts and skills acquired during their technical training in systems and networks.
The student must submit a personal project for approval. It is also possible for him to carry out a research project.
The graduation project is presented orally and in the form of a professional caliber technical report. The project can be carried out in single, pair or trinomial.
This project must demonstrate mastery of all the knowledge acquired in the study program in question and the ability to adapt knowledge to a specific and real case. In doing so, the student must demonstrate autonomy and initiative. In all cases, the work must be carried out in accordance with the regulations for end-of-study projects.
This work must be done relatively independently; the teacher guides the students, provides possible solutions, without specifying the entire process. It is during this course that students fulfill the requirements of the program summary test according to procedures to be specified during the session.

Objective of the subject

At the end of the project, the student will be able to demonstrate his ability to:

- design elements, systems and processes that correspond to specific needs;
- design a work plan and timelines using project planning documents;
- formulate a problem, find information relevant to the solution of the problem raised in the problem, analyze this information, synthesize it, design and state recommendations;
- write a professional-calibre technical report including: the problem, the objectives, the hypotheses, the review of the documentation, the methodology, the design process, the presentation of the results and their analysis, the conclusions and the recommendations.
- The student must devote, on average, six hours per week (for a total of 180 120 hours) to the activities necessary for the realization of the study or the project, the drafting of reports and the preparation of his oral presentation.
- He will have discussion sessions with his supervisor during the session in order to receive advice, report on the progress made, receive feedback on the various deliverables that have been requested of him, etc.
- When in teams, students will perform tasks according to roles that will be determined during project planning

Team work

- Students carry out a graduation project based on a subject of common interest.
- A teacher-supervisor is appointed to ensure the general supervision of the team.

This project is designed to allow students to synthesize the knowledge acquired during the two years of teaching, while getting closer to the context of the company. The objective is therefore to allow the student to be operational very quickly and to facilitate his professional integration. The interest of such an organization is to oblige the students to acquire a methodology of group work.

Similarly, the objectives of the project are to introduce the group to research work; use their technical and conceptual skills in the experimental study of complex problems in systems

and networks; develop practice, exercise autonomy in the laboratory in understanding problems and interpreting results; communicate effectively in writing and orally. This project can be bibliographic and/or laboratory work.

Activities aimed at perfecting the training of students through the acquisition of new knowledge and the observation of working methods and techniques used in the systems and network laboratories and more particularly in the fields of application of the program. Projects will be crowned with a public oral presentation as well as a written report, both activities forming part of the student assessment.

The main objective of the project is creating a network system in a real world scenario. This should include the topics covered during TS1 and TS2, and in particular include network design, implementation, automation, security and virtualization.

The steps to follow are:

1 Participate in the development of the project:
   - 1.1 Feasibility study.
   - 1.2 Drafting specifications.
   - 1.3 Development of technical files.

2 Participate in the decision and the choice:
   - 2.1 From the wiring method.
   - 2.2 Equipment.
   - 2.3 The operating system on each station (Two separate operating systems must exist on the stations)

3 Participate in the follow-up of the works:
   - 3.1 Conformity of the cabling to the specifications.
   - 3.2 Acceptance of the site (if possible)

4 Install network hardware and software
   - 4.1 Indispensable participation in the preparation of the server (hardware and software) and the installation of network cards on the stations.
   - 4.2 Regularly: installation of applications on the network.
   - 4.3 Installation of users and their access rights.

5 Network Maintenance
   - 5.1 Update of the network monitoring book.

6 Network protection:
   - 6.1 Server Protection
     - 6.1.1 Power cut protection.
   - 6.2 Network security:
     - 6.1.2 Essential discriminating access rights. (Not everyone has access to just anything.)
     - 6.1.3 Confidentiality ensured by a password specific to each user.
     - 6.1.4 Essential installation of a network anti-virus.

7 Network resources must be easily usable:
   - 7.1 Installation of a print server allowing simplified print management. (If applicable)
   - 7.2 Installation of an automatic backup on the network. (SAN, NAS, RAID, etc.)

8 Backup Planning:

8.1. Create a detailed backup plan that establishes when backups should be performed, their frequency and retention time.
8.2. Server Configuration: Configure both servers (Windows and Linux) to perform data backups to a cloud service. Integrated operating system backup tools or third-party software can be used.
8.3. Choosing a Cloud Provider: Identify a service provider that offers storage and backup services. Describe your choice based on factors such as reliability, security, cost and features offered.

9 Programming and scripting automation :
9.1. Create scripts to automate network configuration tasks, such as setting up IP addresses, configuring firewalls, and managing network services.
9.2. Utilize scripting skills to develop custom scripts for monitoring network performance and security. This involves creating scripts that can collect data from various sources, process it, and present it in a useful format or alert administrators to potential issues.
9.3. Scripting for Backup and Recovery: Implement scripting solutions for automating the backup and recovery process. This includes writing scripts that can automatically perform backups according to the schedule defined in the backup plan, verify the integrity of backups, and automate the recovery process when needed.
9.4. Use scripting to enhance network security by automating the deployment of security policies, scanning for vulnerabilities, and managing security updates.

Documentation

The thesis document must contain the following parts:

- Cover page.
- Summary.
- Foreword.
- Study of the existing: Positions, activities, physical plan, reviews, solution.
- Physical topology: cabling, distance, topology, equipment
- Logical Topology: Addressing and Topology
- Security: VLAN, VPN, Access list, Firewall, IPD, IDS,… (if applicable)
- Redundancy: STP, RSTP, SAN, NAS, … (if applicable)
- Configuration: Operating system, routers, switch, ACL, ISA,…
- Use of a network simulator: Packet Tracer, NetSim, Ns-3, etc.
- Conclusion.
- Appendices (screens, reports, programs).
- Bibliography.

The size of the project implies the constitution of a team made up of a maximum of three students.
Each group of students must submit a written report and a digital support (if applicable).
The style to be used in the realization of the memory is obligatorily the Font Times New Roman, size 12.
The pagination of the document is mandatory.
The document must contain at least 50 pages without annexes.

INTERNSHIP (60 Periods)

Training internship in a IT company

The internship is a period of practical training and professional experimentation that students will undertake in a company to acquire specific IT and/or Network sector skills, apply the knowledge gained during their academic studies, and gain direct experience in their field of interest. Characterized by a practical and field-oriented approach, an internship aims to integrate academic training with practical experience in a real working environment. During the internship, students will have the opportunity to work on concrete projects, collaborate with industry professionals, and develop cross-functional and soft skills.

1.1 Internship Objectives:

- Development of Technical Skills: Acquire a practical knowledge in the field of study through the practical application of concepts learned in the classroom.
- Development of Professional Skills: improve skills necessary to operate effectively in a professional environment.
- Reinforcement of Soft Skills: enhance soft skills such as leadership, problem-solving, and adaptability, essential for success in a professional career.
- Networking and Professional Connections: establish professional relationships with colleagues, supervisors, and other industry professionals to foster future job opportunities.
- Career Guidance: gain practical experience in the field of interest, allowing interns to confirm or redirect their professional choices.
- Development of Autonomy and Initiative: promote independence and the ability to take initiative in tackling new tasks and problem-solving.

1.2 Internship Tools:

The internship agreement is a formal agreement between three parties: the student (intern), the educational institution (school, university, or training institute), and the host organization (company, organization, or entity) that welcomes the student for a period of practical training. This document establishes the terms and conditions of the internship experience, providing a legal and organizational framework for the activities carried out during the internship period.

Internship Agreement: is an essential tool to ensure clarity, transparency, and compliance with regulations during the internship experience, protecting the interests of all parties involved.

The internship agreement typically includes the following elements:

- Student Information
- School Information and Responsible Contact Person
- Host Company Information
- Ethical and Behavioral Norms
- Monitoring and Evaluation Procedures
- Signatures of the Involved Parties (School, Student, Company)

Internship Training Plan: is a key tool to ensure a structured learning experience consistent with the student's educational and professional objectives. Its drafting requires effective collaboration among all parties involved. The internship training plan is a document outlining the objectives, activities, and implementation methods of a period of

practical training within an organization. This document is usually prepared in collaboration between the student, the school, and the hosting company.

The training plan should include:

- Educational Objectives: a detailed description of the objectives the student should achieve during the internship.
- Contents and Activities: a list of specific activities the student will perform during the internship.
- Duration and Schedule:
- Tutoring and Supervision: appointment of a company tutor or supervisor responsible for supporting and guiding the student during the internship.
- Evaluation Methods: description of how the student's performance will be assessed.
- Obligations and Responsibilities of the Involved Parties: clear definition of the obligations and responsibilities of the student, the school, and the hosting company during the internship period.
- Ethical and Behavioral Norms: ethical guidelines and behavioral norms that the student must follow during the internship.
- Resources and Tools: a list of resources, tools, and access that the student will have available to carry out the activities outlined in the training plan.

Evaluation tools for an internship are devices used to measure, assess, and document a student's performance and progress during the internship period. These tools are designed to provide accurate and constructive feedback on the internship experience and to evaluate the achievement of educational objectives.
Typically, the following are used:

- Evaluation Sheets filled out by the hosting company to assess technical, behavioral, and cross-functional skills acquired by the student.
- Self-assessment Questionnaire that allows the student to reflect on their performance.
-

Final Student Report: is a written document summarizing the student's experience during the internship period. This report provides an overall view of the activities carried out, the skills acquired, the results achieved, and personal reflections on lessons learned.
Here is a possible format and content for a final intern report:

- Introduction: a brief presentation of the student, indicating name, course of study, and educational institution.
- Context and Purpose of the Internship: a description of the hosting company and the context in which the internship took place.
- Activities Carried Out
- Skills Acquired:
- Challenges Faced: a discussion of the challenges and obstacles encountered during the internship.
- Feedback from the Company Tutor and Colleagues:
- Personal Reflections
- Integration with the Educational Path: a discussion of how the internship experience integrates with the student's academic path.
- Suggestions and Recommendations
- Conclusions

THIRD YEAR

SPECIALIZATION SUBJECTS

1. COMPUTER TECHNOLOGY III
2. PROGRAMMING III
3. PROGRAMMING THE WEB III
4. NETWORKS AND SECURITY PROTOCOLS III
5. MAINTENANCE METHODOLOGY
6. BASIC OF CLOUD COMPUTING AND AI II
7. STORING DATA II
8. FINAL REPORT

COMPUTER TECHNOLOGY III
(30 PERIODS OF PRACTICAL EXERCISES)

OBJECTIVE

The aim of the course is to understand what a board with a microcontroller is, how it works, and its main applications. Students will learn the basic syntax of the board with a microcontroller programming language and how to use it to control hardware (serial communication and sensors). Throughout the course, students will have the opportunity to apply their knowledge through practical exercises. At the end of the course, the class will be divided into groups, and each group will have the opportunity to complete a project that will be presented to the rest of the class.

Remarks: this subject could be part of the official exam

CHAPTER 1 (2 Periods of practical exercises)

Introduction to programmable board with a microcontroller (e.g. Aduino)

1.1. Course presentation and objectives.
1.2. Introduction to a Board with a microcontroller: what it is and its applications.
1.3. Description of the Board with a microcontroller development environment (IDE).
1.4. Installation and configuration of the Board with a microcontroller IDE.
1.5. Introduction to the Board with a microcontroller programming language: basic code structure.

CHAPTER 2 (3 Periods of practical exercises)

Fundamentals of Electronics (Review)

2.1 Basic concepts of electronics: voltage, current, resistance.
2.2 Ohm's law and resistance calculation.
2.3 Basic electronic components: resistors, LEDs, buttons, sensors.
2.4 Electric circuits and component connections.

CHAPTER 3 (5 Periods of practical exercises)

Board with a microcontroller Programming

3.1 Basic structure of a Board with a microcontroller program: setup() and loop().
3.2 Use of digital and analog pins.
3.3 Basic commands: pinMode(), digitalWrite(), digitalRead(), analogRead(), analogWrite().
3.4 Time control with delay() and millis().
3.5 Use of variables and data types.

CHAPTER 4 (5 Periods of practical exercises)

Serial Communication and Sensors

4.1 Introduction to serial communication.
4.2 Use of the serial monitor for debugging and output.
4.3 Reading digital and analog sensors.
4.4 Use of motion sensors and ultrasonic sensors.

CHAPTER 5 (10 Periods of practical exercises)

Practical Exercises

5.1 Exercise 1: Flashing LED.
5.2 Exercise 2: Controlling an LED with a button.
5.3 Exercise 3: Controlling LED brightness with a potentiometer.
5.4 Exercise 4: Reading a temperature sensor.
5.5 Exercise 5: Controlling a servo motor.
5.6 Exercise 6: LCD Display: introduction and usage.
5.7 Exercise 7: Creating a thermometer with an LCD display.
5.8 Exercise 8: Creating a parking sensor with ultrasonic sensors.

CHAPTER 6 (5 Periods of practical exercises)

Final Project

6.1 Group work to develop a complete Board with a microcontroller project.
6.2 Presentation of projects and discussion among groups.

PROGRAMMING III

(90 PERIODS OF WHICH 60 ARE PRACTICAL EXERCISES)

OBJECTIVE

This curriculum advances from Programming II, focusing on integrating Artificial Intelligence with Python and developing web APIs using Flask. Students will explore advanced Python topics, AI fundamentals, and web services development. The course is designed to enhance analytical skills and provide hands-on experience with real-world applications.

Teacher's Notes:

- Continual assessment through quizzes, assignments, and practical exercises.
- End-of-module projects to apply the covered concepts.
- Guest Lectures/Workshops by software development experts (if possible/available).
- Peer Programming: Encourage collaborative learning and peer-to-peer teaching.
- Recommended use of diverse online resources, including advanced tutorials and developer forums for extended learning.

CHAPTER 1 (10 Periods of which 5 are practical exercises)

Recap of Programming II

1.1 Review of Object-Oriented Programming, Data Structures, and Algorithms in Python

1.2 Review of Web Development with Python

CHAPTER 2 (20 Periods of which 15 are practical exercises)

Advanced web development with Python

2.1 Flask: Advanced Features and Patterns

2.2 Building RESTful Web APIs with Flask

2.3 Implementing Authentication and Authorization in Web Applications

2.4 Deploying Python Web Applications (Docker, Cloud Hosting)

CHAPTER 3 (25 Periods of which 15 are practical exercises)

Python and AI

3.1 Python Libraries for AI (SciKit-Learn, TensorFlow, PyTorch)

3.2 Building Basic AI Models (Classification, Regression, Clustering)

3.3 Natural Language Processing (NLP) with Python

3.4 Computer Vision with Python (OpenCV, Image Processing)

3.5 Practical Implementation AI in Python

CHAPTER 4 (15 Periods of which 10 are practical exercises)

Advanced Python Topics

4.1 Asynchronous Programming in Python (Asyncio)
4.2 Performance Optimization and Memory Management
4.3 Working with Python in Big Data (PySpark, Hadoop Integration)

Advanced Scripting and Automation with Python

CHAPTER 5 (5 Periods)

Advanced Database Management

5.1 NoSQL Databases
5.2 Advanced ORM Techniques

CHAPTER 6 (15 Periods of practical exercises)

Final Project

6.1 Development of an Advanced Python Project (Web API/AI Integration)
6.2 Project Presentation and Peer Review
6.3 Feedback Session and Reflective Learning

PROGRAMMING THE WEB III
(60 PERIODS OF PRACTICAL EXERCISES)

OBJECTIVE

Building on the understanding of HTML, CSS, and basic JavaScript from the previous courses, Programming the Web III delves into advanced JavaScript concepts and introduces the React framework. This course is designed to equip students with the skills necessary to develop more complex, interactive web applications. It emphasizes hands-on learning and real-world application of advanced scripting and framework-based development.

CHAPTER 1 (10 Periods of practical exercises)

JavaScript review, asynchronous programming and ES6 features

1.1 Quick Revision of JavaScript
1.2 Asynchronous JavaScript: Promises, Async/Await
1.3 Higher-Order Functions and Callbacks
1.4 JavaScript ES6 Features: Arrow Functions, Template Literals

CHAPTER 2 (10 Periods of practical exercises)

JSON and AJAX

2.1 Introduction to JSON
2.2 Retrieving data with AJAX
2.3 RESTful services and HTTP verbs
2.4 Using the fetch API to get, post, put and delete data.

CHAPTER 3 (5 Periods of practical exercises)

JavaScript classes

3.1 Object-Oriented JavaScript: Classes, Objects, Inheritance
3.2 Modules and Namespaces
3.3 Mixins and Higher-Order Functions
3.4 Private Class Fields: Implementing data privacy in classes

CHAPTER 4 (5 Periods of practical exercises)

Advanced DOM Manipulation

4.1 DOM Manipulation in Depth
4.2 Handling Events: Bubbling and Capturing
4.3 Form Validation and Regular Expressions

CHAPTER 5 (5 Periods of practical exercises)

Introduction to React

5.1 Understanding React
5.2 React vs Angular vs Vue
5.3 React Components and Props
5.4 State and Lifecycle in React

CHAPTER 6 (10 Periods of practical exercises)

Developing apps with React

6.1 Handling Events in React
6.2 Conditional Rendering
6.3 Lists and Keys
6.4 Forms and Controlled Components

CHAPTER 7 (10 Periods of practical exercises)

React in depth

7.1 React Router for Navigation
7.2 State Management with Context API and Redux
7.3 Using External APIs with React
7.4 Hooks: useState, useEffect, Custom Hooks

CHAPTER 8 (5 Periods of practical exercises)

Final Project

7.5 Students will develop a small web app with React.
7.6 Peer reviews and teacher feedback is strongly encouraged.

NETWORKS AND SECURITY PROTOCOLS III
(120 PERIODS OF WHICH 90 ARE PRACTICAL EXERCISES)

CHAPTER 1 (15 Periods)

Network Recap

1.1 Review of Concepts:
- 1.1.2 Networking terms, types, connectors, and models.
- 1.1.3 Protocols, parameters, devices, and topologies.
- 1.1.4 IP addresses, classes, DNS, and DHCP.

1.2 Transmission:
- 1.2.1 Elements, Media and Techniques.

1.3 Cybersecurity:
- 1.2.2 Importance.
- 1.2.3 Threats: (Malware, Phishing, DoS, MITM…).
- 1.2.4 Security Vulnerability: (Hardware, Software, and Updates).
- 1.2.5 Privacy Protection: (Encryption, Multi factor…).
- 1.2.6 Roles and Technologies: (IDS, IPS, VPN…).

CHAPTER 2 (15 Periods)

Network Security:

2.1 Security Types, Benefits, and Challenges:
- 2.1.1 Security types: (Antivirus, Antimalware, Cloud Security, Data Loss, Email, Firewall…).
- 2.1.2 Security Benefits
- 2.1.3 Functionality.
  - Confidentiality, Integrity and Availability.
  - Intellectual property protection.
- 2.1.4 Security Challenges: (Remote access, User adherence…).

CHAPTER 3 (90 Periods of practical exercises)

Final Network Applications Project

Divide Students in Groups:

3.1 Project Objectives and scope:
- 3.1.1 Select a Topic.
- 3.1.2 Goals.
- 3.1.3 Define Scope.

3.2 Research:
- 3.2.1 Network gaps and solutions.
- 3.2.2 Network technologies and applications to be used.

3.3 System Design and Architecture:
- 3.3.1 Design all network system and applications.
- 3.3.2 Identify network protocols will be used.

3.4 Development and Implementation:
- 3.4.1 Develop Server and clients (Backend, Frontend).
- 3.4.2 Implement necessary network features.
- 3.4.3 Test Network applications and Fix Bugs.

3.5 Document, Present and Demonstrate:
- 3.5.1 User and Technical documents:
  - User (Documents using networks).

- Technical (Documents about Architecture, Structure and protocols).
- Present Objectives, Design and Outcomes.
- Show a live Demonstration If it's possible.

3.6 Final Submission and Evaluation:

3.6.1 Submit final project includes all documents.

3.6.2 Instructors evaluate project.

3.6.3 Class Discussion between Groups and Instructor.

3.7 Network Tools and Materials Recap (during the lab):

3.7.1 Usage of tools and Lab Materials.

3.7.2 Power Off equipment, Wear personal protective.

3.7.3 All Lab Material:

- Cables (Types and categories).
- Cabling tools (RJ45, Sleeving, Combs etc.…).
- Cabinets.
- Patch Panels.
- Ethernet Hubs.
- LAN / WAN / WLAN routers.
- Measuring equipment (Reflectometer to test fiber cables, Category 7 cable scanner (bidirectional tests), LAN / WAN Protocol Analyzer)
- Proxy server for internet Access.
- Network Printer.
- Internet connections : WIFI, and Cable.
- ISDN adapter Card (Integrated Services Digital Network).
- Operating systems (Windows 10, Windows Server, Linux open source)
- Network antivirus software
- Network plan Software (Microsoft Visio or else)
- File compression software
- Multiboot boot software (Partition Magic, System Commander.).
- Remote maintenance and control Tool : "PCAnywhere, Vnc, etc."
- Network element monitoring tools : "Openview, WhatsUp Gold, Ciscoview, etc.)
- Internet software (email clients, FTP clients, browsers, videoconferencing, etc.).

Final Network Applications Project examples:

The following are possible examples for the Final Network Applications Project.
Teachers can update or modify the exercises as needed.

Exercise 1: Designing and Implementing a Windows Active Directory Infrastructure

Objective:

To design and implement a network infrastructure using Windows Server technologies, focusing on Active Directory Domain Services (AD DS) for user and computer management.

Scope:

Select a Topic: Implementing Active Directory for a small enterprise.
Identify Goals: To enable centralized management of users, computers, and policies.
Define Scope: The project will involve setting up a domain controller, organizing an OU (Organizational Unit) structure, creating user and computer accounts, and implementing Group Policy Objects (GPOs) for security settings and network configurations.

Tasks:

Research:

- Investigate the best practices for Active Directory architecture.
- Select appropriate network protocols for Active Directory communication.
- Design the structure to reflect the organization's administrative and geographical structure.
- Identify the network protocols to be used within the Active Directory environment (e.g., LDAP, Kerberos).
- Design cabling / infrastructure

Development and Implementation:

- Install Windows Server on a virtual machine and configure it as a domain controller.
- Develop a script to automate the creation of user accounts and computer objects.
- Implement Group Policy Objects to manage security settings and software deployment.

Document, Present, and Demonstrate:

- Prepare user documentation on how to log in to the domain and access network resources.
- Create technical documentation detailing the Active Directory structure, GPOs implemented, and the script for account creation.
- Present the design and outcomes to the class and demonstrate the login process and effect of GPOs on client computers.

Exercise 2: Setting Up a Linux-Based Web Hosting Environment

Objective:

To create a Linux-based web hosting environment using open-source technologies.

Scope:

Select a Topic: Linux web server and database management.

Identify Goals: To host a dynamic website that utilizes a database.

Define Scope: This exercise involves installing and configuring a Linux server with Apache, MySQL, and PHP (LAMP stack), hosting a website, and securing the server.

Tasks:

Research:

- Explore Linux distributions best suited for web hosting.
- Identify network technologies and applications for hosting a dynamic website.
- Design the network architecture for a LAMP stack.
- Choose network protocols for web communication (e.g., HTTP, HTTPS).
- Design cabling / infrastructure

Development and Implementation:

- Install a Linux operating system on a virtual machine.
- Develop and deploy a LAMP stack, including security configurations (firewall settings, SSL certificates).
- Implement a simple dynamic website that interacts with a MySQL (or Oracle) database.

Document, Present, and Demonstrate:

- Create user documentation on how to interact with the website.
- Prepare technical documentation that outlines the server architecture, security measures, and website functionality.
- Present the project to the class, highlighting the design decisions and demonstrating the website functionality.

Lab Material Usage:

For both exercises, students can utilize network cables, cabling tools, and network testing equipment to connect and test the server's physical connectivity.

Network monitoring tools can be employed to observe the traffic to and from the servers, identifying potential bottlenecks or security issues.

MAINTENANCE METHODOLOGY
(90 PERIODS OF WHICH 60 ARE PRATICAL EXPERIENCES)

OBJECTIVE

At the end of this course the student should be able to:

- Use fault finding techniques
- Utilize of online constructive resources (Update Download): drivers, lists of known incidents, etc.
- Employ maintenance documentation
- Read architectural plans
- Plan tasks
- Produce a report
- Apply the skills acquired in a real work context (internship)

CHAPTER 1 (3 Periods)

Solving difficulties by addressing the causes

1.1 Lack of standards, methods and tools
  1.1.1 Design flaws and programming errors
  1.1.2 Defects and insufficient documentation
  1.1.3 Insufficient testing
  1.1.4 Operating stress analysis faults
1.2 Lack of clear and measurable objectives
  1.2.1 Increase in turnover
  1.2.2 Lack of resources and distribution of scarcity
1.3 Technological aging

CHAPTER 2 (10 Periods)

Different maintenance categories

2.1 Corrective maintenance
  2.1.1 Palliative Maintenance
  2.1.2 Curative maintenance
2.2 Adaptive maintenance
2.3 Evolutionary maintenance

CHAPTER 3 (8 Periods)

Framework of interventions

3.1 Incident management: correction of breakdowns (corrective, curative, hot maintenance)
3.2 Change management
  3.2.1 Perfective maintenance
  3.2.2 Adaptive maintenance
  3.2.3 Preventive maintenance
  3.2.4 Evolutionary maintenance
3.3 Project development
3.4 Technical support

CHAPTER 4 (6 Periods)

Follow-up of actions

4.1 The cost of the operation
4.2 How soon can it be done
4.3 Economic interest
4.4 Load measurement and productivity
4.5 Quality control
4.6 Advantages of Third Party Maintenance

CHAPTER 5 (3 Periods)

Production of a report

CHAPTER 6 (60 Periods of practical exercise)

Training internship in a IT company

The internship is a period of practical training and professional experimentation that students will undertake in a company to acquire specific IT sector skills, apply the knowledge gained during their academic studies, and gain direct experience in their field of interest. Characterized by a practical and field-oriented approach, an internship aims to integrate academic training with practical experience in a real working environment. During the internship, students will have the opportunity to work on concrete projects, collaborate with industry professionals, and develop cross-functional and soft skills.

6.1 Internship Objectives:

- Development of Technical Skills: Acquire a practical knowledge in the field of study through the practical application of concepts learned in the classroom.
- Development of Professional Skills: improve skills necessary to operate effectively in a professional environment.
- Reinforcement of Soft Skills: enhance soft skills such as leadership, problem-solving, and adaptability, essential for success in a professional career.
- Networking and Professional Connections: establish professional relationships with colleagues, supervisors, and other industry professionals to foster future job opportunities.
- Career Guidance: gain practical experience in the field of interest, allowing interns to confirm or redirect their professional choices.
- Development of Autonomy and Initiative: promote independence and the ability to take initiative in tackling new tasks and problem-solving.

6.2 Internship Tools:

The internship agreement is a formal agreement between three parties: the student (intern), the educational institution (school, university, or training institute), and the host organization (company, organization, or entity) that welcomes the student for a period of practical training. This document establishes the terms and conditions of the internship experience, providing a legal and organizational framework for the activities carried out during the internship period.

Internship Agreement: is an essential tool to ensure clarity, transparency, and compliance with regulations during the internship experience, protecting the interests of all parties involved.

The internship agreement typically includes the following elements:

- Student Information

- School Information and Responsible Contact Person
- Host Company Information
- Ethical and Behavioral Norms
- Monitoring and Evaluation Procedures
- Signatures of the Involved Parties (School, Student, Company)

Internship Training Plan: is a key tool to ensure a structured learning experience consistent with the student's educational and professional objectives. Its drafting requires effective collaboration among all parties involved. The internship training plan is a document outlining the objectives, activities, and implementation methods of a period of practical training within an organization. This document is usually prepared in collaboration between the student, the school, and the hosting company.

The training plan should include:

- Educational Objectives: a detailed description of the objectives the student should achieve during the internship.
- Contents and Activities: a list of specific activities the student will perform during the internship.
- Duration and Schedule:
- Tutoring and Supervision: appointment of a company tutor or supervisor responsible for supporting and guiding the student during the internship.
- Evaluation Methods: description of how the student's performance will be assessed.
- Obligations and Responsibilities of the Involved Parties: clear definition of the obligations and responsibilities of the student, the school, and the hosting company during the internship period.
- Ethical and Behavioral Norms: ethical guidelines and behavioral norms that the student must follow during the internship.
- Resources and Tools: a list of resources, tools, and access that the student will have available to carry out the activities outlined in the training plan.

Evaluation tools for an internship are devices used to measure, assess, and document a student's performance and progress during the internship period. These tools are designed to provide accurate and constructive feedback on the internship experience and to evaluate the achievement of educational objectives.

Typically, the following are used:

- Evaluation Sheets filled out by the hosting company to assess technical, behavioral, and cross-functional skills acquired by the student.
- Self-assessment Questionnaire that allows the student to reflect on their performance.
-

Final Student Report: is a written document summarizing the student's experience during the internship period. This report provides an overall view of the activities carried out, the skills acquired, the results achieved, and personal reflections on lessons learned.

Here is a possible format and content for a final intern report:

- Introduction: a brief presentation of the student, indicating name, course of study, and educational institution.
- Context and Purpose of the Internship: a description of the hosting company and the context in which the internship took place.
- Activities Carried Out
- Skills Acquired:
- Challenges Faced: a discussion of the challenges and obstacles encountered during the internship.

- Feedback from the Company Tutor and Colleagues:
- Personal Reflections
- Integration with the Educational Path: a discussion of how the internship experience integrates with the student's academic path.
- Suggestions and Recommendations
- Conclusions

BASICS OF CLOUD COMPUTING AND AI II

(90 PERIODS OF WHICH 60 ARE PRACTICAL EXERCISES)

CHAPTER 1 (25 Periods of which 15 are practical exercises)

Networking and Cloud Infrastructure

- 1.1 Cloud Networking
  - 1.1.1 Virtual networks, VPNs, and network security in a cloud infrastructure (such as Azure, AWS or GCP).
  - 1.1.2 Hands-on setup of network configurations and security groups.
- 1.2 Cloud Storage and Database Solutions
  - 1.2.1 Advanced concepts in cloud storage and database management.
  - 1.2.2 Hands-on with SQL and NoSQL databases in the cloud.
- 1.3 Server Solutions in the Cloud
  - 1.3.1 Exploring different server solutions: virtual machines, dedicated hosts, and containers.
  - 1.3.2 Setting up and managing cloud servers for different scenarios.
- 1.4 Load Balancing and Traffic Management
  - 1.4.1 Techniques for managing load balancing and traffic in cloud environments.
  - 1.4.2 Implementing load balancers and CDN services in a cloud setting.

CHAPTER 2 (20 Periods of which 15 are practical exercises)

Hybrid and Multi-Cloud Environments

- 2.1 Hybrid Cloud Concepts
  - 2.1.1 Understanding hybrid cloud models and their real-world applications.
  - 2.1.2 Building a hybrid cloud setup using tools like Azure Arc, AWS Outposts, and Google Anthos.
- 2.2 Cloud Migration Strategies
  - 2.2.1 Planning and executing cloud migration.
  - 2.2.2 Practical migration of an application from on-premises to the cloud.
- 2.3 Cloud Security and Compliance
  - 2.3.1 Cloud security best practices and tools.
  - 2.3.2 Implementing and managing compliance in a multi-cloud or hybrid environment.

CHAPTER 3 (25 Periods of which 15 are practical exercises)

Artificial Intelligence and Machine Learning

- 3.1 Review of Fundamentals of Machine Learning and Machine Learning in Practice
  - 3.1.1 Key concepts of machine learning
  - 3.1.2 Deep learning for computer vision and natural language processing.
- 3.2 Natural Language Processing (NLP) Techniques
  - 3.2.1 Sentiment analysis, text classification, and language generation.
  - 3.2.2 Hands-on project: Create an NLP model to perform a specific task like chatbot creation.
- 3.3 Ethics and Future of AI
  - 3.3.1 Discussing the ethical implications and future challenges in the field of AI.
  - 3.3.2 Group discussion or presentation on the potential future developments and societal impacts of AI.
- 3.4 AI for Predictive Analysis and Decision Making
  - 3.4.1 Understanding the use of AI for predictive analytics in various sectors.

3.4.2 Practical exercises and case studies on implementing AI for business intelligence.

CHAPTER 4 (20 Periods of which 15 are practical exercises)

Applied AI Projects and Cloud Integration

4.1 AI Project Design and Development
- 4.1.1 Planning and developing an AI project.
- 4.1.2 Group projects: Identify a problem and create an AI solution.

4.2 Cloud-based AI Solutions
- 4.2.1 Integrating AI models with cloud platforms.
- 4.2.2 Deploying AI applications in the cloud.

4.3 Industry-Specific AI Applications
- 4.3.1 Exploring AI applications in various sectors like healthcare, finance, and retail.
- 4.3.2 Case studies of AI success stories in different industries.

4.4 Presentation and Project Demonstration
- 4.4.1 Effective presentation skills for AI projects.
- 4.4.2 Final demonstration of group projects.

STORING DATA II

(60 PERIODS OF WHICH 30 ARE PRACTICAL EXERCISES)

CHAPTER 1 (6 periods of which 3 of practical exercises)

Storing Data SQL Recap:

1.1 SQL Overview, Services, Schemas, Data types, SSMS.
1.2 Sort & filter, JOINs, Subqueries, Built-in functions.
1.3 Database principles (Design, Create…), Security (Permissions, schemas…)

CHAPTER 2 (6 periods of which 3 of practical exercises)

Introduction to Oracle Databases:

2.1 Overview of Oracle Database.
2.2 Oracle key Features.
2.3 Differences Between SQL Server and Oracle.
2.4 Differences Between T-SQL and PL/SQL.
2.5 Installation of Oracle Database software.

CHAPTER 3 (6 periods of which 3 of practical exercises)

Introduction to PL/SQL

3.1 Introduction
3.2 Explore the structure of PL/SQL statements
3.3 Examine the SELECT and WHERE statements
3.4 Work and compare Columns, Characters and Rows
3.5 Exercise - Work with SELECT and WHERE statements

CHAPTER 4 (6 periods of which 3 of practical exercises)

Order by and Row functions in PL/SQL

4.1 Logical Comparison and procedure rules
4.2 Sorting Rows
4.3 Introduction to Functions
    4.3.1 Case and Character manipulation.
    4.3.2 Number, Date, Null Functions.
    4.3.3 Conditional Expressions.
4.4 Exercise – Order by, Sorting and Function types.

CHAPTER 5 (8 periods of which 4 of practical exercises)

Combine multiple tables with JOINs in PL/SQL

5.1 Understand joins concepts and syntax
5.2 Use inner joins
5.3 Use outer joins
5.4 Use cross joins
5.5 Use Natural joins
5.6 Self Joins and Hierarchical Queries.
5.7 Exercise - Query multiple tables with joins

CHAPTER 6 (6 periods of which 3 of practical exercises)

Use Group functions in PL/SQL

6.1 Categorize Group functions
6.2 Use COUNT, DISTINCT, and NVL
6.3 Use group functions and Clauses.
6.4 Use Rollup, Cube Operations, and Group Sets.
6.5 Using Set Operators
6.6 Exercise - Use Group functions

CHAPTER 7 (6 periods of which 3 of practical exercises)

Write Subqueries in PL/SQL

7.1 Subqueries fundamentals
7.2 Use single-row and multiple-row subqueries
7.3 Use correlated subqueries
7.4 Exercise - Use subqueries

CHAPTER 8 (6 periods of which 3 of practical exercises)

Identify DML and DDL

8.1 Define Data Manipulation Language
8.2 Define Data Definition Language
8.3 Work with DML commands (Insert, Update, Delete, select)
8.4 Work with DDL commands (Create, Alter, Drop)
8.5 Examine the MERGE and Multi-Tables
8.6 Exercise - Work with DML and DDL Commands

CHAPTER 9 (2 periods of which 1 of practical exercises)

Constraints Overview

9.1 Intro to Constraints (NOT NULL and UNIQUE)
9.2 Examine PRIMARY KEY, FOREIGN KEY, and CHECK Constraints
9.3 Managing Constraints.

CHAPTER 10 (8 periods of which 4 of practical exercises)

Client Connections to Databases:

10.1 Introduction to Database Connectivity with Oracle Databases
10.2 Connection Methods and Protocols
10.3 Configuring Client Software for Database Access
10.4 Security Considerations in Database Connections
10.5 Exercise - Setting Up and Testing Database Connections

FINAL REPORT

The final report, which the student will need to compile at the end of the year, should contain the following points:

CHAPTER 1

Final training internship

The final training internship report should contain a series of essential points that reflect the experience and learning of the entire training internship period. Here are some points that could be included:

1.1 Introduction and Contextualization:
- 1.1.1 Presentation of the company or organization where the training internship was done.
- 1.1.2 Contextualization of the project or task assigned during the training internship .

1.2 Training internship Objectives:
- 1.2.1 Description of the objectives set at the beginning of the training internship.
- 1.2.2 Definition of the specific objectives of the assigned project.

1.3 Activities Undertaken:
- 1.3.1 Detailed list of activities and tasks done during the training internship.
- 1.3.2 Description of the methodologies and tools used to complete the assigned tasks.

1.4 Learning and Skills Acquired:
- 1.4.1 Discussion of the technical, professional, and soft skills acquired during the training internship.
- 1.4.2 Analysis of the challenges encountered and the solutions adopted to overcome them.
- 1.4.3 Reflections on the evolution of one's skills in the work environment.

1.5 Feedback and Evaluations:
- 1.5.1 Receipt of feedback and periodic evaluations from the company tutor or supervisor.
- 1.5.2 Personal reflections on the effectiveness of the feedback received and improvements made.

1.6 Conclusions and Recommendations:
- 1.6.1 Summary of the experiences and results obtained during the training internship.
- 1.6.2 Recommendations for improving the training internship process or for future similar activities.

1.7 Attachments:
- 1.7.1 Any documents, reports, multimedia material produced during the training internship.

CHAPTER 2

Web Computing Project Report

The report on a web computing project should include several key points reflecting the design, development, and implementation process of the web system.

2.1 Introduction to Web Computing Project
- 2.1.1 A general overview of the project and the context in which it was developed, including the objectives and motivations underlying the project.

2.2 System Design
- 2.2.1 Architecture of the web system.
- 2.2.2 Description of the technologies and programming languages used in the system design.

2.3 Development and Implementation
- 2.3.1 Description of decisions made during development and solutions adopted to address any encountered issues.

2.4 System Functionality and Features
- 2.4.1 Illustration of the key features of the created web system and their functionalities.

2.5 Testing and Validation
- 2.5.1 Results of tests conducted and any issues encountered during the testing phase.

2.6 Links
- 2.6.1 Links's website created and a portion of code as an example of the work done.

CHAPTER 3

Programming Project (Python) Report

A report on a programming project with Python should describe the design, development, and implementation process of the software. The report should include the following points:

3.1 Introduction to the Project
- 3.1.1 A general overview of the project, including objectives and the context in which the software was developed.

3.2 Requirements Analysis
- 3.2.1 Detailed description of the functional and non-functional requirements of the developed software.

3.3 Software Design
- 3.3.1 Architecture of the software, including modules, classes, and interfaces.
- 3.3.2 Description of the technologies and frameworks used in the software design.
- 3.3.3 UML diagrams or other tools used to visualize the software architecture and design.

3.4 Development and Implementation
- 3.4.1 Description of the software development process, including coding, testing, and debugging phases.
- 3.4.2 Explanation of decisions made during development and solutions adopted to address any encountered issues.

3.5 Software Functionality and Features
- 3.5.1 Detailed description of the implemented software functionalities.
- 3.5.2 Illustration of the key features of the software and their functionalities.

3.6 Testing and Validation
- 3.6.1 Description of tests conducted on the software to ensure correct functionality and compliance with requirements.

3.7 Code Documentation

3.7.1 Detailed documentation of the source code, including code comments.

3.8 Performance Considerations and Optimization

3.8.1 Analysis of software performance, including execution times and resource usage.

3.9 Conclusions and Future Perspectives

3.9.1 Considerations on the work done, results obtained, and description of any elements for improving the developed software.